NPS72-87-002CR

# NAVAL POSTGRADUATE SCHOOL

## Monterey, California

CONTRACTOR REPORT

TRANSIENT REACTIVE EXHAUST FLOW FROM

A RING-SYMMETRIC HF/DF SPACE LASER

by

Joseph Falcovitz

March 1987

Approved for public release; distribution unlimited.

Prepared for: Strategic Defense Initiatve Office
The Pentagon
Washington, DC 20301-7100

NAVAL POSTGRADUATE SCHOOL
Monterey, California

RADM R. C. Austin
Superintendent

D. A. Schrady
Provost

The work reported herein was performed for the Naval Postgraduate School by Dr. Joseph Falcovitz under contract N62271-86-M-0214. The work presented in this report is in support of "Rarefied Gas Dynamics of Laser Exhaust Plume" sponsored by the Strategic Defense Initiative Office/Directed Energy Office. This is the final report for that contract. The work provides information concerning transient reactive flow in the laser diffuser section. The project at the Naval Postgraduate School is under the cognizance of Distinguished Professor A. E. Fuhs who is principal investigator.



Prepared by:

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE



# REPORT DOCUMENTATION PAGE

| | |
|---|---|
| 1a REPORT SECURITY CLASSIFICATION: UNCLASSIFIED | 1b RESTRICTIVE MARKINGS: NONE |
| 2a SECURITY CLASSIFICATION AUTHORITY | 3 DISTRIBUTION/AVAILABILITY OF REPORT: Approved for Public Release; Distribution Unlimited |
| 2b DECLASSIFICATION/DOWNGRADING SCHEDULE | |
| 4 PERFORMING ORGANIZATION REPORT NUMBER(S): NPS72-87-002CR | 5 MONITORING ORGANIZATION REPORT NUMBER(S): NPS72-87-002CR |
| 6a NAME OF PERFORMING ORGANIZATION: JOSEPH FALCOVITZ; 6b OFFICE SYMBOL (If applicable): 72 | 7a NAME OF MONITORING ORGANIZATION: NAVAL POSTGRADUATE SCHOOL, Code 72 |
| 6c ADDRESS (City, State, and ZIP Code): Research Contractor, Naval Postgraduate School, Code 72, Monterey, CA 93943-5100 | 7b ADDRESS (City, State, and ZIP Code): Space Systems Academic Group, Monterey, CA 93943-5100 |
| 8a NAME OF FUNDING/SPONSORING ORGANIZATION: Strategic Defense Initiative Office; 8b OFFICE SYMBOL (If applicable): SDIO/DEO | 9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER |
| 8c ADDRESS (City, State, and ZIP Code): SDIO/DEO, Washington, DC 20301-7100 | 10 SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO; PROJECT NO; TASK NO; WORK UNIT ACCESSION NO |

11 TITLE (Include Security Classification)

TRANSIENT REACTIVE EXHAUST FLOW FROM A RING-SYMMETRIC HF/DF SPACE LASER

12 PERSONAL AUTHOR(S)
JOSEPH FALCOVITZ

| 13a TYPE OF REPORT | 13b TIME COVERED | 14 DATE OF REPORT (Year, Month, Day) | 15 PAGE COUNT |
|---|---|---|---|
| Contractor Report | FROM JAN 87 TO MAR 87 | March 1987 | 51 |

16 SUPPLEMENTARY NOTATION

| 17 COSATI CODES | | | 18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number) |
|---|---|---|---|
| FIELD | GROUP | SUB-GROUP | Laser Exhaust, Spacecraft Contamination, Chemical Laser, Exhaust Plume, Hydrogen Recombination, Transient Flow |

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

Transient diffuser flow in an HF/DF laser is analyzed via a simplified model. A radial laser configuration is assumed where the diffuser is idealized as a ring-symmetric channel of uniform width. The outward radial flow following an abrupt start-up of diffuser inflow is computed numerically using a GRP-type code for integrating the Euler equations, which include an ongoing reaction of hydrogen recombination. A new kinetic scattering effect is suggested, resulting from the recoil of an HF or DF molecule assisting as third body in the hydrogen recombination reaction. The main conclusion is that transient diffuser plume flow effects do not drastically aggravate the potential for contaminating backflow (HF+DF) caused by either thermal or kinetic backscattering.

| | |
|---|---|
| 20 DISTRIBUTION/AVAILABILITY OF ABSTRACT: ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS | 21 ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED |
| 22a NAME OF RESPONSIBLE INDIVIDUAL: ALLEN E. FUHS, Distinguished Professor | 22b TELEPHONE (Include Area Code): (408)646-2948; 22c OFFICE SYMBOL: Code 72 |

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

☆ U.S. Government Printing Office: 1986—606-24.

## ABSTRACT

Transient diffuser flow in an HF/DF laser is analyzed via a simplified model. A radial laser configuration is assumed where the diffuser is idealized as a ring-symmetric channel of uniform width. The outward radial flow following an abrupt start-up of diffuser inflow is computed numerically using a GRP-type code for integrating the Euler equations, which include an ongoing reaction of hydrogen recombination. A new kinetic scattering effect is suggested, resulting from the recoil of an HF or DF molecule assisting as third body in the hydrogen recombination reaction. The main conclusion is that transient diffuser/plume flow effects do not drastically aggravate the potential for contaminating backflow (HF + DF) caused by either thermal or kinetic backscattering.

## ACKNOWLEDGEMENT

This work was conducted as part of a laser exhaust study under the cognizance of Distinguished Professor Allen E. Fuhs. I deeply appreciate and wish to thank Professor Fuhs for his continuous support and guidance.

# TABLE OF CONTENTS



# LIST OF FIGURES

## NOMENCLATURE followed by units (if any)

| | |
|---|---|
| $A$ | cross-section area (of diffuser or nozzle) |
| $A_v$ | Avogadro's number $6.022 \times 10^{26}$ (molecules/kmole) |
| $C$ | sound speed (m sec$^{-1}$) |
| $E$ | total energy per unit volume (MJ m$^{-3}$) |
| $f_X$ | mole fraction of species X at diffuser entrance |
| $g_c$ | rate of hydrogen recombination collisions (with HF or DF) per unit volume (m$^{-3}$ sec$^{-1}$) |
| $k_0$ | hydrogen recombination rate constant [(mole/cm$^3$)$^{-2}$ sec$^{-1}$] |
| $k_1$ | modified hydrogen recombination rate constant [(kg/m$^3$)$^{-2}$ sec$^{-1}$] |
| $M$ | Mach number |
| $n$ | number density (molecules/m$^3$) |
| $P$ | pressure (Pa) |
| $Q_H$ | formation energy of $H_2$ (KJ/mole) |
| $Q_{DET}$ | hydrogen recombination energy release parameter (MJ/kg) |
| $t$ | time (ms) |
| $T$ | temperature (K) |
| $U$ | flow velocity (m/sec) |
| $W$ | average molecular weight (kg/kmole) |
| $x$ | radial coordinate (in diffuser) |
| $Z$ | normalized mole fraction of H ($Z = 1$ at diffuser entrance) |
| $\alpha$ | turning angle of flow velocity in a corner expansion to vacuum |
| $\Delta\alpha$ | turning angle increment for transient corner expansion |
| $\gamma$ | specific-heat ratio |
| $\rho$ | density (kg m$^{-3}$) |
| $\theta$ | inclination of flow velocity vector (to x-axis) |
| $\Gamma$ | the fraction $(\gamma + 1)/(\gamma - 1)$ |
| $[X]$ | molar density of species X (kmole m$^{-3}$) |

INDICES

| | |
|---|---|
| $(\ )_0$ | stagnation conditions |
| $(\ )_1$ | nozzle (diffuser) exit conditions |
| $(\ )_2$ | diffuser entry conditions |
| $(\ )_t$ | Eulerian time derivative |
| $(\ )_x$ | Eulerian space derivative |
| $\dot{(\ )}$ | Lagrangian time derivative |

# 1. INTRODUCTION

In recent years a study of the gasdynamics of space-based HF/DF laser exhaust plume was conducted at the NPS. The aim of this program was to identify phenomena that give rise to a potentially contaminating backflow of corrosive species (HF, DF, F), and to estimate the magnitude of the flux arriving at the spacecraft [1, 2, 3, 4] . The laser was envisioned as a cylindrical spacecraft having a centrally located nozzle of ring-symmetry. This is an idealization of a more general zero-thrust exhaust configuration for an open loop HF/DF laser (Fig. 1-1).

In former studies [2, 3] we focused on the contribution of thermal self-scattering to a contaminating backscattered flux of corrosive molecules (HF, DF) from the exhaust plume. It was shown that this flux emanates primarily from the lip-centered rarefaction fans that flank the exhaust plume, and it was estimated that at a sufficiently high exit Mach number (e.g. $M_1 = 4$), the thermally backscattered flux of HF + DF is utterly negligible.

The purpose of this report is to present a study of the effect of diffuser start-up flow, including the ongoing recombination of atomic hydrogen, on the thermally backscattered flux. Our main conclusion is that by designing for a sufficiently high steady exit Mach number (e.g. $M_1 = 4$), the level of thermally backscattered flux of corrosive molecules (HF + DF) would be negligible even when the combined effects of hydrogen recombination and transient flow are considered.

The diffuser is simplified as a cylindrically expanding channel of uniform width, and the flow is idealized as inviscid expansion of a perfect gas with an ongoing reaction of hydrogen recombination. The flow is computed as one-dimensional time-dependent.

A fully 2-D time-dependent computation of an emerging exhaust plume is outside the scope of the present laser exhaust study. Rather, we resort to semi-quantitative arguments that lead to an upper-bound estimate of the effect of transient flow on thermal backscattering, by showing that the transient turning angle in an expansive corner flow can be estimated as higher than the corresponding steady flow turning angle. It is subsequently suggested that an overestimate to the the backscattered flux from a transient plume is obtained by considering it as a steady plume whose exit velocity vector is pre-rotated so that its limiting (vacuum) streamline coincides with that defined by the transient turning angle.

The plan of this report is the following. In Ch. 2 we present our radial (1-D) diffuser model based on the linear configuration of the TRW test laser [5] , and we propose a "typical case" of HF/DF laser flow based on one of those tests. Ch. 3 is devoted to the gasdynamic governing equations, including the hydrogen recombination rate. The resulting diffuser model was implemented in a 1-D Euler code (named HFL) which utilizes the GRP scheme for integrating the conservation laws of compressible flow [6] . The HFL code is given in Appendix A. The results of a typical diffuser start-up flow are presented in Ch. 4, along with a discussion and analysis of the effects of transient flow and uncertainty in the hydrogen recombination rate on thermally backscattered flux of HF + DF. We also consider kinetic backscattering of HF or DF molecules caused by third-body recoil in the hydrogen recombination reaction. It is shown that the combined contribution of these effects to backscattered flux remains negligible in the typical case. Chapter 4 ends with some concluding remarks, and references are brought in Ch. 5.

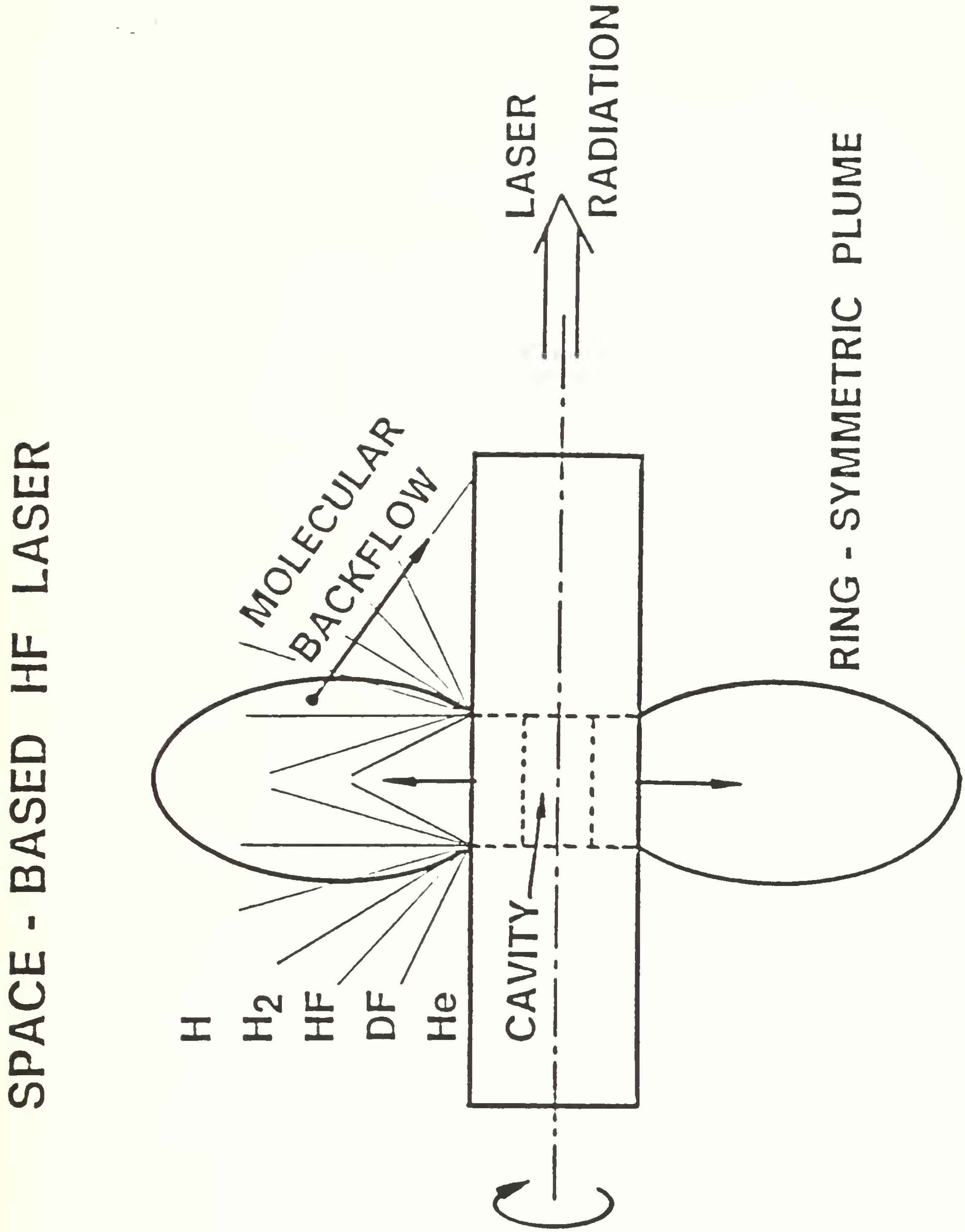


Figure 1-1. Ring-Symmetric HF DF Laser Exhaust Plume

## 2. THE DIFFUSER FLOW MODEL

Our diffuser model is based on a schematic radial laser configuration as shown in Fig. 2-1. The major components in this design concept are those present in an experimental HF/DF open loop laser tested at TRW [5] . In this chapter we describe the radial configuration including the diffuser (Section 2.1), then we present a typical HF/DF laser flow based on one of those TRW tests (Section 2.2).

### 2.1 Radial Laser Configuration

We assume that an open loop HF/DF laser of the type tested at TRW [5] , can be rearranged in a radial (rather than linear) sequence, where the flow begins at the hub and proceeds in an outward radial direction. Referring to Fig. 2-1. the major components of this configuration are :

(a) Combustion Chamber : Deuterium and fluorine burn with excess fluorine, resulting in hot gaseous mixture where fluorine is virtually completely dissociated.

(b) Nozzle Cascade : Rapid expansion to supersonic flow leaves atomic fluorine concentration effectively frozen.

(c) $\mathbf{H_2 + He}$ Injection : Mixture is injected between the supersonic streams of $\mathbf{DF + F}$ emerging from the cascade.

(d) Mixing and Lasing : The lasing is from vibrationally excited **HF** molecules produced by direct reaction between $\mathbf{H_2}$ and **F**. As a by-product, one **H** atom is produced for every **HF** molecule.

(e) Diffuser Entrance : This point marks the end of the lasing process. From this point on the flow is just an exhaust to be discarded safely.

(f) Radial Diffuser : The purpose of the diffuser is to raise the flow Mach number at the exit so that no appreciable backflow from the exhaust plume will take place [2, 3, 4] . It should be noted that a desirable diffuser area ratio can be achieved at lower radius ratios by letting the diffuser expand in the axial direction.

### 2.2 Typical Laser Flow

We chose as a typical laser flow one of the tests conducted in the TRW experiments [5] it is test III in Table 5 (p. 91) of that reference. The mole fraction and corresponding average molecular weight and specific-heat ratio (**W** and $\gamma$) at the diffuser entrance for this test were :

Mole fractions $f_H = .091 \quad f_{HF} = .091 \quad f_{H_2} = .104$

$f_{DF} = .135 \quad f_{He} = .579$

(2.2-1)

Average molecular weight $W = 7.27$

Specific-heat ratio $\gamma = 1.54$

The additional data required for the computational model is the flow variables at the diffuser entrance. Some of these variables have been measured directly or indirectly and the remaining variables can be evaluated from standard isentropic flow relations for compressible flow of an ideal gas. The data given in the TRW tests report [5] are :

Diffuser entrance cross-section is : 1″ × 7″

Mass flow rate is : 14.99 (gr $sec^{-1}$)

Stagnation temperature is : 1400 (K) (2.2-2)

Flow velocity is : 2300 (m $sec^{-1}$)

Flow density is now obtained as the ratio of the specific mass flow rate and the velocity. Then Mach number is extracted from the standard relations between Mach number, velocity and stagnation temperature. The values of these variables are :

$$M_2 = 2.26$$

$$\rho_2 = 1.44 \times 10^{-3} \quad (kg\ m^{-3}) \tag{2.2-3}$$

$$P_2 = 9.72 \times 10^{-4} \quad (MPa)$$

The flow at the diffuser entrance is thus fully specified for the selected typical case. In the next chapter we take up the matter of computing the transient expansion following an abrupt start-up of the diffuser inflow.

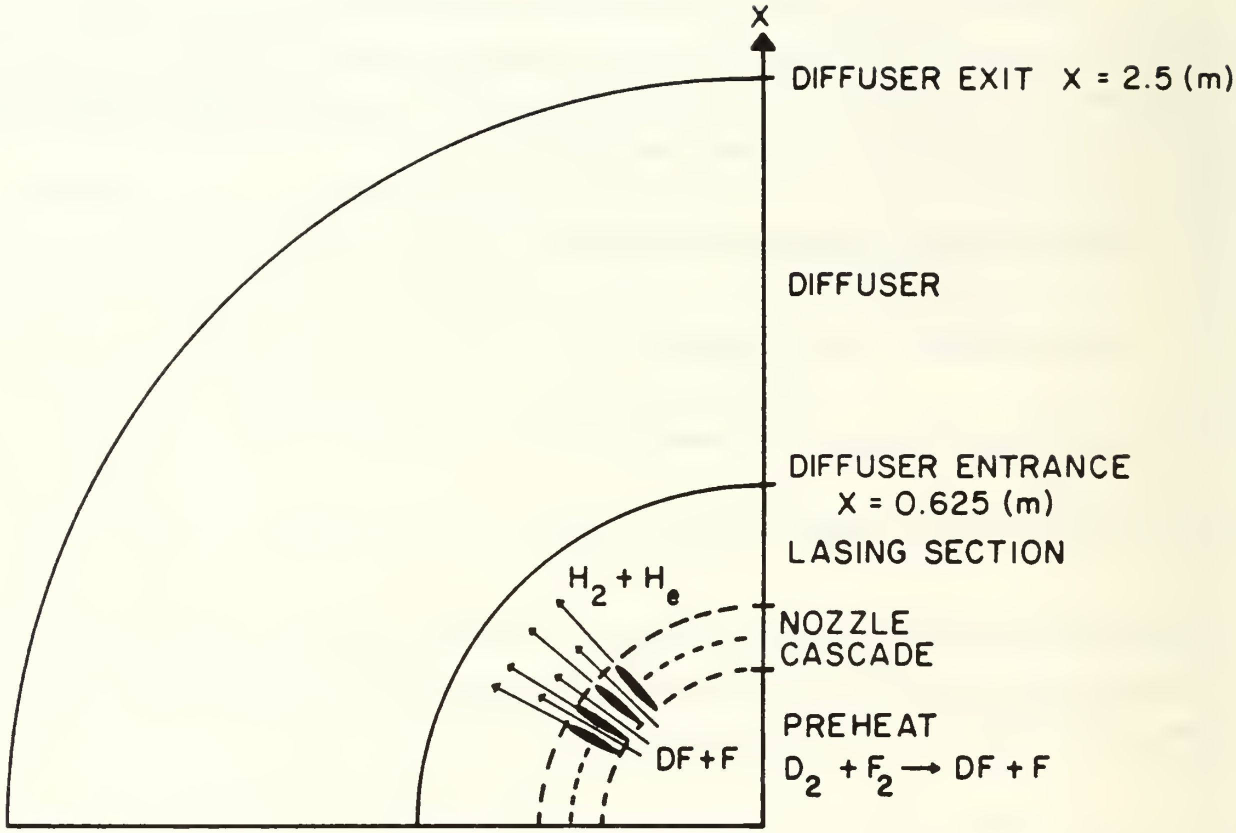


Figure 2-1. Radial Configuration of HF/DF Open-Loop Laser

## 3. THE GASDYNAMIC FLOW MODEL

The transient expansion of the lasing products following an abrupt start-up of inflow at the diffuser entrance, is idealized as an inviscid compressible flow of a mixture of perfect gases with one ongoing chemical reaction – that of hydrogen recombination. The computational model thus calls for a formulation of the governing equations and for a numerical scheme capable of integrating them in time and space.

In this chapter we describe the governing equations for the diffuser flow (Section 3.1), following by a simple approximation chosen for the hydrogen recombination rate as a three body reaction (Section 3.2). The numerical scheme employed for the solution is of the Generalized Riemann Problem (GRP) type [6] . This scheme has been implemented in a code which was adapted to the diffuser flow case. The code (named HFL) and some brief description of features introduced to treat hydrogen recombination are given in Appendix A. In a recent report [7] a very similar version of this code (without chemical reaction) was described in considerable detail.

### 3.1 The Governing Equations

The expansion of lasing products through the diffuser is governed by the Euler equations for a gaseous mixture of ideal gases with ongoing hydrogen recombination reaction. We write these equations in the quasi-one-dimensional format of flow in a stream tube of varying cross-section area $A(x)$, but in actual computations we set $A(x)=x$, thereby reducing the equations to the cylindrical case. (The code HFL, however, can accept any smooth $A(x)$).

We chose to simplify the designation of atomic hydrogen mole fraction by normalizing it as $f_H Z(x,t)$, where $f_H$ is the mole fraction at the diffuser entrance (it is constant) and $Z(x,t)$ denotes the degree of dissociation ($Z=1$ is maximum concentration of H at the diffuser entrance, $Z=0$ is complete recombination). As an approximation, we assume that $W$ and $\gamma$ are constant throughout. In the typical case (2.2-1), $\gamma$ and $W$ can vary by as much as 1% and 4% respectively at full recombination. Neglecting that variation is consistent with the preliminary nature of the present diffuser flow analysis.

The governing equations are the three standard conservation laws (mass, momentum and energy), and an additional species conservation law for atomic hydrogen. The conservation laws are augmented by an equation of state (ideal gas) and by a rate law for hydrogen recombination. The full system of equations is :

| | | |
|---|---|---|
| Mass | $[A\rho]_t + [A\rho U]_x = 0$ | |
| Species H | $[A\rho Z]_t + [A\rho UZ]_x = A\rho\dot{Z}$ | |
| Momentum | $[A\rho U]_t + [\rho U^2]_x + AP_x = 0$ | (3.1-1) |
| Energy | $[AE]_t + [A(E+P)U]_x = 0$ | |
| Equation of State | $P(\rho,E,Z,) = (\gamma - 1)\,[E - \rho U^2/2 - (\rho f_H Z/W)Q_H]$ | |
| Recombination Rate | $\dot{Z} = \dot{Z}(\rho,P,Z)$ | |

where A is a function solely of x, and ρ, U, P, E, Z are all functions of (x,t). The normalized rate function is $\dot{Z}$. Its derivation and explicit expression are given in Section 3.2.

We note that the energy equation contains the heat released by hydrogen recombination ($Q_H$ per mole of H) in an implicit way, by defining the total energy as including the latent heat due to recombination, in addition to internal and kinetic energy terms.

### 3.2 Hydrogen Recombination

The hydrogen recombination is commonly assumed to be a three-body reaction [8] , with the following rate law :

$$H + H + M \rightarrow H_2 + M$$

$$[\dot{H}_2] = -k_0\,[H]^2\,[M] \qquad (3.2\text{-}1)$$

$$k_0 = 7 \times 10^{15}\ [(\text{mole/cm}^3)^{-2}\ \text{sec}^{-1}]$$

where by M we denote a third molecule, which in our case may be any molecule in the gaseous mixture (including H).

The value we chose for $k_0$ is an upper limit of a range of values recommended by Kondratiev [9] for the reaction H + H + He (from $2 \times 10^{15}$ to $7 \times 10^{15}$), since in the typical case helium constitutes about 58% of the molecules in the lasing products.

The data compiled by Kondratiev [9] indicates a comparable range of variation with temperature as with the third species M. We later introduce uncertainty factors of 10 and 100, which in all likelihood more than reflect the uncertainty in the reaction rate for hydrogen recombination.

It is convenient to redefine the reaction rate in terms of flow density $\rho$ and normalized H concentration Z. The modified rate constant $k_1$ is given by :

$$\dot{Z} = - k_1 \rho^2 Z^2$$

$$k_1 = 2 k_0 f_H / W^2 \qquad (3.2\text{-}2)$$

where the factor 2 is due to the fact that the rate of depletion of H is twice the rate of production of $H_2$. In deriving (3.2-2) we made use of the relations $f_H = [H]_2/[M]_2$ and $[M] = \rho/W$, where index 2 denotes the diffuser entrance.

The heat of formation released per mole of recombined $H_2$ is 435.783 (kJ/mole) according to [10] (page F-179 in that reference). In the code we use the energy parameter per unit mass of the gaseous mixture $Q_{DET}$, defined as :

$$Q_{DET} = Q_H f_H / W$$

$$Q_H = 435.783 / 2 \quad (\text{KJ/mole}) \qquad (3.2\text{-}3)$$

## 4. RESULTS AND DISCUSSION

Several numerical computations of the transient diffuser flow were performed using the code HFL (Appendix A). The flow was started by an abrupt inflow at the diffuser inlet; the radial diffuser model and the typical HF/DF laser case (Sections 2.1 and 2.2) were assumed. Due to uncertainty in hydrogen recombination rate, three rate levels were assumed : the nominal level (3.2-1), a tenfold increased rate and a hundredfold increased rate.

We are primarily concerned with the thermally backscattered flux arriving at the surface from the exhaust plume. Our goal is to show that by choosing a diffuser with a sufficiently high steady exit Mach number, the backscattered flux can be negligibly low even when the combined effects of hydrogen recombination and transient exit flow are taken into account. Put in other words, we present a simplified analysis of the transient exhaust flow, which enables the establishment of a reasonable margin on a frozen/steady design for low backscattered flux. This analysis is presented in Section 4.1 below.

The continuation of the hydrogen recombination reaction into the plume may give rise to a contaminating backflow of a different kind. The velocity imparted to the "third body" molecule in the hydrogen recombination process (Eq. 3.2-1) may be large enough to overcome the radial flow velocity component, resulting in molecular backflow. This effect is discussed in Section 4.2, and it is shown to be potentially capable to give rise to an HF/DF deposition rate of the order of 1 molecular monolayer per hour.

### 4.1 Thermal Backscattering

Since in steady flow thermally backscattered flux is linked primarily to exit Mach number, we focus our attention on the time-history of the exit Mach number obtained from the diffuser start-up computations mentioned above. Two major features are noted in these results (Fig. 4-1).

The first feature is the monotonic decrease in exit Mach number with time, as the flow within the diffuser undergoes transition from an initial "cloud expansion" mode to a steady expansion flow in a channel of increasing cross-section area. A steady exit flow seems to be established after about 2 (ms) from the start-up instant. The second feature is related to the heat released in the flow by hydrogen recombination. As the recombination rate is increased, so does the time-asymptotic value of the exit recombination fraction. The recombination fractions in the three cases (Fig. 4-1) are 1%, 7% and 46%; the corresponding exit Mach numbers are 4.0, 3.69 and 2.81. This trend is in qualitative agreement with the well known phenomenon of decrease in Mach number as a result of heat addition to a steady supersonic flow in a channel of uniform cross-section [11] .

Estimates of recombination rates are notoriously uncertain. How can a reasonable uncertainty factor be established? We do not know of an estimate of this factor for the flow of HF/DF laser products. However, in another case of flow involving hydrogen recombination - that of hydrazine rocket motors - a recent study [12] recommends a hydrogen recombination rate of $2.8 \times 10^{17}/T$ (in c.g.s. K units) and an uncertainty factor of 30. Assuming the relatively low temperature of $T = 300$ (K) and multiplying by 30, we get a recombination rate of $3 \times 10^{16}$, versus $7 \times 10^{16}$ in our tenfold case. This demonstrates that the tenfold case is already reasonably high; just the same, we also retain the hundredfold case in the upcoming discussion.

The exit Mach number time-histories (Fig. 4-1) demonstrate that the nominal diffuser flow is nearly frozen (exit recombination fraction 1% and Mach number 4); even in the tenfold case the flow is not dominated by hydrogen recombination (exit recombination fraction 7% and Mach number 3.69). This conclusion is in agreement with findings of the TRW HF/DF laser study [5] . It takes the unrealistically high hundredfold increase in hydrogen recombination rate to produce a decisive change in the steady diffuser flow (exit recombination fraction 46% and Mach number 2.81).

One role of the diffuser is to expand the laser exhaust to a sufficiently high exit Mach number, so that the thermally backscattered flux of corrosive molecules (HF, DF) is negligibly small. Let us consider the nominal case ($M_1 = 4$) and denote by "reference flux" the combined HF+DF backscattered flux arriving at a point located at 0.1 (m) from the nozzle lip (Fig. 1-1). Using the code RINGBD based on the breakdown surface model [2] , the nominal reference flux was evaluated as $3.1 \times 10^5$ (molecules $m^{-2}$ $sec^{-1}$). This flux level corresponds to a surface deposition rate (assuming the sticking coefficient equals 1) of about $10^{-10}$ molecular monolayers per hour, which is utterly negligible (a reasonable estimate of total operating time would not exceed several hours).

Observing that a steady exit flow has been established in all three cases by about $t = 2$ (ms) (see Fig. 4-1), we suggest that an approximate estimate of the backscattered flux would be obtained by assuming a steady flow at the diffuser exit and the exhaust plume. The RINGBD computations of the increase hydrogen recombination rate cases yielded the following results. In the tenfold case ($M_1 = 3.69$) the flux was $10^2$ times larger than the nominal reference flux. In the hundredfold case

($M_1 = 2.81$) the flux increase was by a factor of $10^7$. Even in the unrealistically high hundredfold case, that flux level corresponds to about $10^{-3}$ monolayers per hour, which is still negligible.

What other uncertainties could affect the foregoing conclusion? First we consider the effect of variations in stagnation temperature and density relative to the nominal case, then we take up the matter of transient plume flow.

Exit stagnation temperature was observed to vary in the HFL diffuser start-up computations by a factor of no more than about 1.3. Since all velocities in the breakdown surface model are normalized by speed of sound [2] , the flux is proportional to the square root of the stagnation temperature. Also, it has been shown [2] that a change in stagnation density can produce an inversely proportional change in flux. Stagnation density was reduced by a factor of about 1.3 in the tenfold case and about 2.8 in the hundredfold case. In either case the potential effect on flux is small relative to changes brought upon by reduced exit Mach number, so we attribute the drastic change in flux following an increased hydrogen recombination rate primarily to the reduced exit Mach number.

We now address the effect of transient plume flow. Since the exit Mach number is monotonically decreasing with time (Fig. 4-1), and since other flow variables do not appreciably affect the flux from a steady plume, we suggest that an upper bound on the effect of transient plume flow can be established by linking it to a "transient turning angle", larger than that corresponding to a steady flow with the same exit Mach number. Obviously, if a Prandtl-Meyer flow pattern is derived from a flow which exits the nozzle at an angle lower than 90°, its limiting turning angle would bring it closer to the spacecraft surface, and with a lower outward radial velocity to overcome, more molecules would be thermally backscattered. Thus, we regard the transient flow around the nozzle lip as represented by an equal exit Mach number steady flow, with some initial exit turning angle.

The justification for this model is the following "upper bound" estimate for the transient turning angle of an emerging exhaust plume. The exit velocity vector $U_1$ is visualized as rotating by a gradual turning of the plume/vacuum interface until it reaches the normal velocity $(2/(\gamma-1))C_1$ (Fig. 4-2), which corresponds to a plane-wave expansion into vacuum. The corresponding turning angle is $\alpha = (2/(\gamma-1))C_1/U_1 = (2/(\gamma-1))/M_1$. It is an overestimate of the turning angle since in a steady corner expansion (and also in a nonsteady one) the magnitude of the velocity vector increases as it rotates, in order to conserve stagnation enthalpy (or even increase it in nonsteady flow about an expansive corner).

It is of interest to note that $\alpha$ as given above is identical with the first term of a power series expansion in $1/M_1$ of the standard Prandtl-Meyer expression for the limiting turning angle [13] . Let us denote by $\Delta\alpha$ the increment of the unsteady turning angle relative to the corresponding steady one (Prandtl-Meyer). This (positive) increment is given by :

$$\Delta\alpha = (2/(\gamma-1))/M_1 - \left\{ \Gamma^{1/2}\arctan\left[\Gamma^{1/2}\left(M_1{}^2 - 1\right)^{-1/2}\right] - \arctan\left[\left(M_1{}^2 - 1\right)^{-1/2}\right] \right\}$$

(4.1-1)

$$\Gamma = (\gamma + 1)/(\gamma - 1)$$

We now argue that the flux ratio between a steady flow and the same flow with exit velocity rotated by $\Delta\alpha$ is an upper-bound estimate to the effect of transient plume flow with the same exit variables. We note that by pre-rotating the steady corner flow, we make the limiting streamline in the steady case, coincide with the nonsteady plume/vacuum interface deemed to have rotated through the angle $\alpha$.

The stage is now set to estimate the transient flux increase factors in the three hydrogen recombination rate cases (exit Mach numbers 4, 3.69 and 2.81). Using Eq.(4-1) we get the turning angle increments of $\Delta\alpha =$ 4.1°, 5.1° and 10.6° respectively. Re-computing the reference flux (code RINGBD) in these cases with initial exit angle of $90^\circ - \Delta\alpha$, we get the flux increase factors of 2.7, 3.5 and 10 respectively. Even in the worst case (hundredfold), the flux would increase from $10^{-3}$ to $10^{-2}$ monolayers per hour, which is still negligibly small.

We also notice that the flux increase factor resulting from the decrease in exit Mach number due to higher hydrogen recombination rate, is much larger than the factor representing the effect of transient plume flow in the same case. Thus, in the tenfold case these factors are $10^2$ versus 3.5, and in the hundredfold case the ratio is even higher : $10^7$ versus 10. We conclude that the transient effect is quantitatively secondary to the effect of exit Mach number, which is thus established as the major parameter for designing an exhaust with a negligible level of thermally backscattered HF + DF flux.

## 4.2 Kinetic Backscattering

The role of the third body in the hydrogen recombination reaction (3.2-1) is to absorb the energy released by the recombination process while maintaining the combined pre-collision momentum of the participating molecules. Assume the momentum added to **[M]** is **q**, then the momentum added to $H_2$ is $-\mathbf{q}$. Denote by $m_1$ the mass of $H_2$, by $m_2$ the mass of **HF** and by **e** the energy released due to the recombination. Then **q** is determined from the following energy conservation relation :

$$q^2/2m_1 + q^2/2m_2 = e$$

$$m_1 = 2 \qquad m_2 = 20 \qquad e = 4.36 \times 10^8 \text{ (Joules/kmole)} \tag{4.2-1}$$

$$U = q/m_2 = 2000 \text{ (m/sec)}$$

The values used in the computation above are per kmole, but the result is the same as using values pertaining to single molecules. Since the velocity obtained upon expansion to zero-pressure in the typical case is about 3000 (m/sec), our results imply that a turning angle in excess of $\arccos(2/3) = 48°$ is needed in order to enable some kinetically backscattered molecules to reach the spacecraft.

In the nominal case the turning angle is 49° and in the tenfold case it is 52°. Also, some kinetically backscattered molecules can originate from within the rarefaction fan, since the exit velocity in the typical case is only about 1500 ($m\ sec^{-1}$) (versus 2000 ($m\ sec^{-1}$) velocity increment to kinetically scattered HF molecules). Consequently, this effect may contribute to spacecraft contamination, and its magnitude should be estimated. It will become progressively more significant at lower exit Mach number, and thus may be an additional factor in determining a minimal value of this flow variable.

While we do not presently attempt at an exact integration of the flux of kinetically backscattered molecules arriving at each surface point, we propose the following overestimate of its magnitude. Consider the outgoing flux of kinetically scattered HF molecules from a half-space of quiescent uniform gas having the thermodynamic properties of the nozzle exit in the typical case. Consider the hydrogen recombination rate equation (3.2-1), and substitute $[X] = f_X n/A_v$. This equation then reads as the volume rate of hydrogen recombinations (denoted as $g_c$ below). In a slab of quiescent gas, half the kinetically scattered molecules have an outward pointing velocity vector in one direction. The probability of collisionless passage out of the slab is $\exp(-x/\lambda)$, so that the outgoing flux **F** is given by :

$$F = g_c \lambda/2$$

$$g_c = k_0 n^3 A_v^{-2} f_H^2 (f_{HF} + f_{DF}) \qquad (4.2\text{-}2)$$

Using typical case exit conditions $n_1 = 2.81 \times 10^{22}$ ($m^{-3}$) and mean free path $\lambda_1 = 1.28 \times 10^{-4}$ (m), we get $F = 5.14 \times 10^{16}$ ($m^{-2}$ $sec^{-1}$), or about 12 monolayers per hour. Since only a small fraction of this flux actually arrives at spacecraft surface points, the arriving flux is in all likelihood well below 1 monolayer per hour. If this flux level is deemed significant, an accurate integration of kinetically scattered flux emanating from the exhaust plume should be performed. The details of such scheme are analogous to the first-collision ambient scattering model [4] , since both effects result in a source-like distribution throughout the plume, and both involve a steric factor (one half in the quiescent gas above) related to the vector addition of flow velocity and velocity imparted through ambient collision or assistance in hydrogen recombination. Therefore, the two effects can conveniently be unified under a common framework.

## 4.3 Concluding Remarks

From the foregoing analysis and discussion we conclude that a design for low (negligible) level of thermally backscattered flux of HF+DF can be accomplished by assuming steady frozen exhaust flow. The effects of transient diffuser/plume flow and hydrogen recombination may typically increase that flux by a factor of $10^2$. Given the sensitivity of flux to exit Mach number, an adequate margin can readily be achieved by raising the exit Mach number. In the typical case presented here, an exit Mach number of 4 has been shown to be adequate.

It should be noted that transient effects originating from points upstream of the diffuser entrance were not considered here, as they depend on details of the system construction and operating sequence.

The kinetic scattering effect has been pointed out as an additional potential source of contaminating backflow. It is the result of hydrogen recombination collisions assisted by an HF or DF molecule as third body. The HF/DF molecule receives part of the kinetic energy release by the hydrogen recombination.

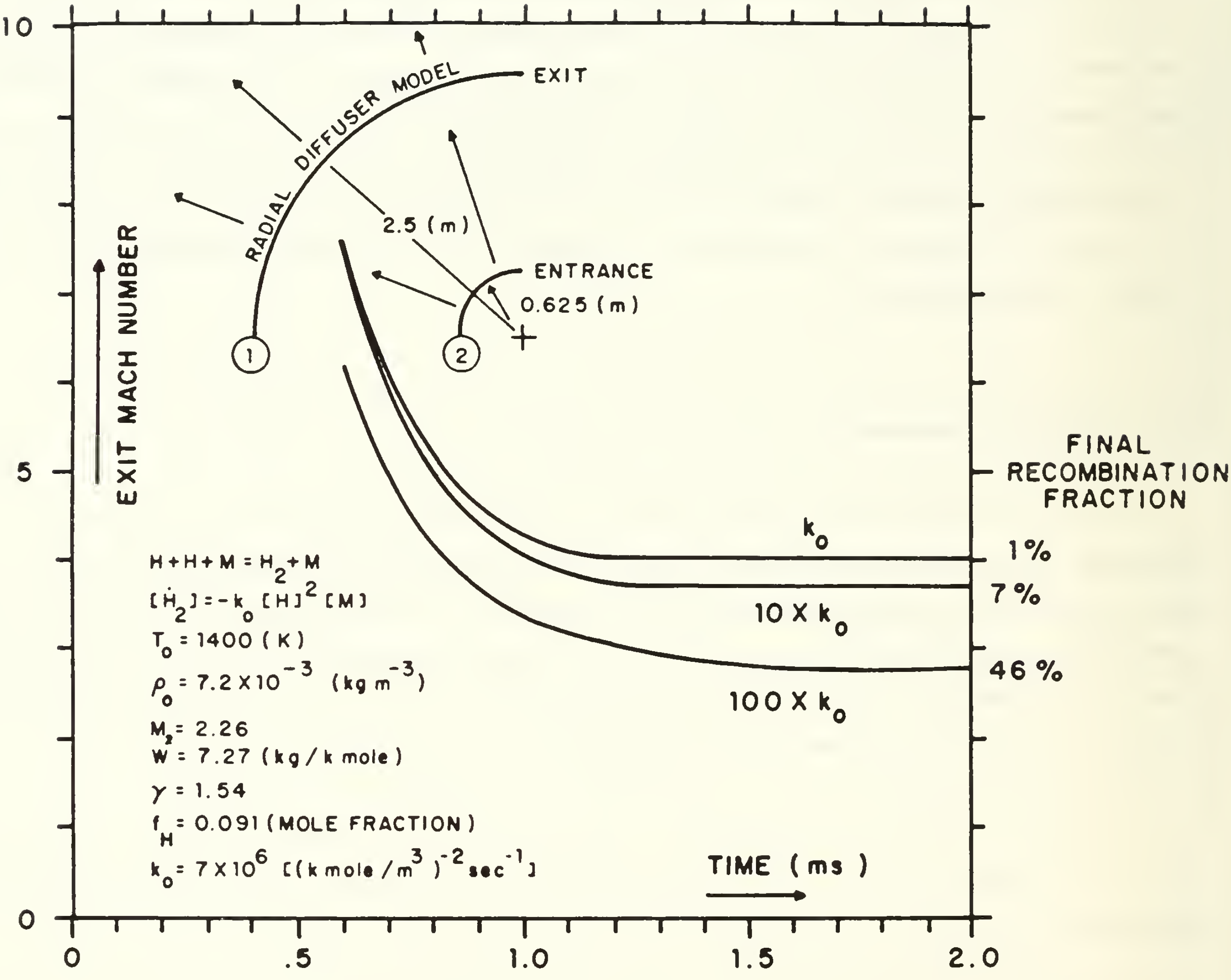


Figure 4-1. Mach Number at Diffuser Exit for Reactive Transient Flow

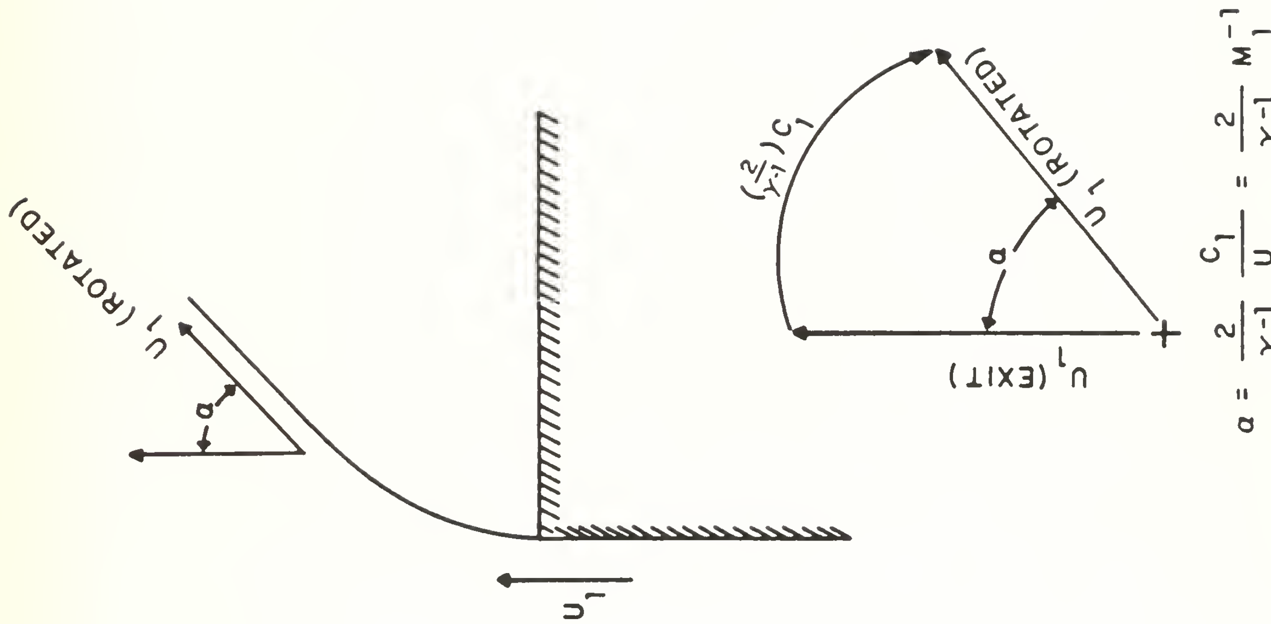


Figure 4-2. Rotation Angle for Non-Steady Plume (Overestimate)

## 5. REFERENCES


[1] Abramovich, S., "Gas Dynamics of Laser Exhaust External to Spacecraft", Naval Postgraduate School, Monterey, CA 93943. Report NPS67-84-006CR, Nov. 1985.

[2] Falcovitz, J., "A Breakdown Surface Model for Thermal Backscattering from the Exhaust Plume of a Space-Based HF Laser", Naval Postgraduate School, Monterey, CA, Report NPS67-86-002CR, June 1986.

[3] Falcovitz, J. and Fuhs, A. E., "An Integral Model for Thermal Backscattering from the Exhaust Plume of a Space-Based HF Laser", AIAA/ASME 4$^{th}$ Joint Thermophysics and Heat Transfer Conference, June 2-4,1986, Boston, Massachusetts. Paper AIAA-86-1320.

[4] Falcovitz, J. and Fuhs, A. E., "Contaminating Sideflow from Supersonic Exhaust Jets in Space", 37$^{th}$ IAF Congress, Oct. 4-11,1986, Innsbruck, Austria. Paper IAF-86-29.

[5] Mastrup, F., Broadwell, E., Miller, J. and Jacobs, T. A., "Hydrogen Fluoride Laser Technology Study", Technical Report AFWL-TR-72-28, October 1972.

[6] Ben-Artzi, M., and Falcovitz, J., "An Upwind Second-Order Scheme for Compressible Duct Flows", ***SIAM Journal on Scientific and Statistical Computing***, Vol 7, p.744-768, 1986.

[7] Falcovitz, J. "Impulsive Loading from a Bare Explosive Charge in Space", Report NPS72-86-004CR, Dec. 1986, Naval Postgraduate School, Monterey, CA 93943.

[8] Sykes, A. G., ***Kinetics of Inorganic Reactions***, Pergamon Press, Oxford, 1966.

[9] Kondratiev, V. N., "Rate Constants of Gas Phase Reactions Reference Book", (English Translation), a publication from the Office of Standard Reference Data, National Bureau of Standards, U.S. Department of Commerce, COM-72-10014, 1972.

[10] Chemical Rubber Company, ***CRC Handbook of Chemistry and Physics***, 64th edition, CRC Press, Florida, 1983.

[11] Shapiro, A. H., ***The Dynamics and Thermodynamics of Compressible Fluid Flow***, The Ronald Press Company, New York, 1953.

[12] Miller, J. M., Hoffman, R. J. and Segal, A., "Chemical Species and Reactions of Importance to Plume Contamination Production", 16th JANNAF Plume Technology Meeting, Sept. 9, 1986, USAF Academy, Colorado Springs, Colorado.

[13] Liepmann, H. W. and Roshko, A., ***Elements of Gasdynamics***, John Wiley, New York, 1957.

## APPENDIX A. The HFL Code

### A.1 Features Specific to Diffuser Flow Model

The purpose of this appendix is to provide the listing of the HFL code used in the diffuser flow computations. An almost identical code was recently described in some detail [7] , the major modification for HFL being the addition of hydrogen recombination.

The highlights of the HFL version to the GRP code [7] are as follows (numbers in parenthesis refer to statement numbers which are prefixed by HFL in the listing).

(a) Data (NETUNM) :

The data which relates specifically to the diffuser flow with hydrogen recombination is given in statements (194 - 239). Some parameters are stored in /DIFFUS/ (129) for use in BEGIN and SAFAE.

(b) Initial Conditions (BEGIN) :

The initial conditions are ideally an empty diffuser. As an approximation we set small pressure and density (270 - 271) and a positive velocity (277) as initial values. The velocity reduces the intensity of the shock driven into the dilute initial gas by the abrupt start-up of inflow at the diffuser entrance (I = 2). Note that the initial value of the total energy E(I) is augmented by the heat of formation of hydrogen (306). The heat term is subsequently subtracted from the total energy in order to compute the pressure (see (d) below).

The grid is defined as follows. The number of cells (L-2) is L-2 = 50 (24). The diffuser extends from X0 = 0.625 to X1 = 3.125 (283 - 284), so that DX = 0.05. Since the external radius of the spacecraft is just 2.5, this grid consists of 40 cells within the diffuser and 10 cells in an extended segment. The purpose of this extended segment is to minimize any propagation of error due to imperfection of the outflow boundary condition (see (c) below). Since the ambient pressure is zero, this additional segment does not affect the flow at $X \leq 2.5$. Consequently, the results for the diffuser exit flow were read from cell I = 39 whose midpoint is X = 2.5.

(c) Boundary Conditions (SAFAE) :

The inflow is set by assigning the diffuser entrance conditions to cell I = 1. The outflow is set by extrapolating (with zero gradient) the flow from the inner boundary cell (I = L-1) to the boundary cell I = L.

(d) Conservation Laws (CYCEUL) :

Here all four conservation laws are integrated in time. There is an added equation of a "time-split" scheme for solving the conservation of **H** species equation. It is first solved without the recombination rate term (459), then the change in Z(I) due to recombination during that time step is added as DZZ (501 - 502).

The energy equation includes the contribution of the formation energy indirectly. This equation has the same format as the adiabatic one (491), but when QDET.GT.0, the internal energy (505) is affected by changes in Z(I) and as a result the pressure P(I) (516) is also affected by progress in the recombination reaction.

A.2 Code Listing

```
C$OPTIONS LIST
      IMPLICIT REAL*8(A-H,O-Z,$)
C  PROGRAM DETO
C  HFL --  HF/DF LASER DIFUSER TRANSIENT FLOW.
      COMMON B(52,26)
     1      ,ENDB
      COMMON /AB/A(50)
      EQUIVALENCE (L,A(1)),(LL,A(2)),(T,A(3)),(DT,A(4)),(TMAX,A(5)),
     1            (TMUD,A(6)),(DTMUD,A(7)),(JOB,A(8)),(NERI,A(9)),
     2            (JJJ,A(10)),(KEYMON,A(11)),(NCYC,A(12))
      EQUIVALENCE (COLELA,A(13))
      EQUIVALENCE (LAGEUL,A(14))
      EQUIVALENCE (UGAL,A(15))
      EQUIVALENCE (KEYEK,A(16))
      EQUIVALENCE (NCYCPR,A(17))
      EQUIVALENCE (STAB,A(18)),(DTBA,A(19)),(DTKOD,A(20)),(KDT,A(21))
      COMMON /MONIT/NC14(4),CASEAV(4),NF16(6),
     1               NMONU(4),NMONP(4),NMONG(4),NMONRO(4),NMONZ(4)
C*********************************************************************
C     DO 20 N=1,30
C20   NC14(N)=0
      NMAT=26
C     L=(LOCF(ENDB)-LOCF(B(1,1)))/NMAT
      L=52
      LL=L-1
      NN=NMAT*L
      DO 1 I=1,L
      DO 1 II=1,NMAT
 1    B(I,II)=0.
      CALL MAINO(L,B(1, 1),B(1, 2),B(1, 3),B(1, 4),B(1, 5),
     1           B(1, 6),B(1, 7),B(1, 8),B(1, 9),B(1,10),
     2           B(1,11),B(1,12),B(1,13),B(1,14),B(1,15),
     3           B(1,16),B(1,17),B(1,18),B(1,19),B(1,20),
     4           B(1,21),B(1,22),B(1,23),B(1,24),B(1,25),
     5           B(1,26))
      STOP
      END
      SUBROUTINE MAINO
     1            (L,X,U,P,RO,G,E,DU,DP,DRO,DG,DXSI,MIN,
     2             US,PS,UIDOT,PIDOT,
     *            FIMZ,ZMDOT,
     3             TENA,FIRO,FIM,FIE,GIP,VOL,Z,DZ)
      IMPLICIT REAL*8(A-H,O-Z,$)
      DIMENSION X(L),U(L),P(L),RO(L),G(L),E(L),DU(L),DP(L),DRO(L),
     1          DG(L),DXSI(L),MIN(L),
     2          US(L),PS(L),UIDOT(L),PIDOT(L)
     3         ,TENA(L),FIRO(L),FIM(L),FIE(L)
     4         ,GIP(L),VOL(L),Z(L),DZ(L)
     5         ,FIMZ(L),ZMDOT(L)
      COMMON /AB/A(50)
      EQUIVALENCE (LL,A(2)),(T,A(3)),(DT,A(4)),(TMAX,A(5)),
     1            (TMUD,A(6)),(DTMUD,A(7)),(JOB,A(8)),(NERI,A(9)),
     2            (JJJ,A(10)),(KEYMON,A(11)),(NCYC,A(12))
      EQUIVALENCE (LAGEUL,A(14))
      EQUIVALENCE (NCYCPR,A(17))
      EQUIVALENCE (STAB,A(18)),(DTBA,A(19)),(DTKOD,A(20)),(KDT,A(21))
      COMMON /TOT/AMTOT,ETOT,EKTOT,EPTOT,TENTOT
C*********************************************************************
      T=0.
      NCYC=0
      JJJ=0
      CALL NETUNM
      DELT=DT
      CALL BEGIN
     1            (L,X,U,P,RO,G,E,DU,DP,DRO,DG,DXSI,MIN,
     2             US,PS,UIDOT,PIDOT,
     *            FIMZ,ZMDOT,
     3             TENA,FIRO,FIM,FIE,GIP,VOL,Z,DZ)
      CALL SAFAE
     1            (L,X,U,P,RO,G,E,DU,DP,DRO,DG,DXSI,MIN,
     2             US,PS,UIDOT,PIDOT,
     *            FIMZ,ZMDOT,
```

```
     3              TENA,FIRO,FIM,FIE,GIP,VOL,Z,DZ)
 1    NCYC=NCYC+1
C TIME STEP CONTROL.
      DT=DTBA
      IF(DT.GT.1.1D0*DTKOD.AND.DTKOD.NE.0.) DT=1.1D0*DTKOD
      IF(NCYC.EQ.2) DT=DT/10.
      IF (NCYC.EQ.1) DT=0.
      IF(DT.EQ.0.) GO TO 11
      NHAD=((TMUD-T)/DT-1.D-10)
      IF(NHAD.GE.10) GO TO 11
      DT=(TMUD-T)/DFLOAT(NHAD+1)
 11   CONTINUE
      T=T+DT
      IF((NCYC/NCYCPR)*NCYCPR.NE.NCYC.AND.NCYC.GT.NCYCPR) GO TO 33
      PRINT 10, NCYC,T,DT,KDT
 10   FORMAT(1X,'NCYC=',I4,3X,'T=',D11.4,3X,'DT=',D11.4,3X,'KDT=',I4)
 33   CONTINUE
      DTBA=DTMUD
      KDT=0
      NERI=1
      IF (DABS(T-TMUD).LT.1.D-8) NERI=0
      CALL CYCEUL
     1             (L,X,U,P,RO,G,E,DU,DP,DRO,DG,DXSI,MIN,
     2              US,PS,UIDOT,PIDOT,
     *             FIMZ,ZMDOT,
     3              TENA,FIRO,FIM,FIE,GIP,VOL,Z,DZ)
      CALL SAFAE
     1             (L,X,U,P,RO,G,E,DU,DP,DRO,DG,DXSI,MIN,
     2              US,PS,UIDOT,PIDOT,
     *             FIMZ,ZMDOT,
     3              TENA,FIRO,FIM,FIE,GIP,VOL,Z,DZ)
      IF (NERI.NE.0) GO TO 2
      CALL PRINT
     1             (L,X,U,P,RO,G,E,DU,DP,DRO,DG,DXSI,MIN,
     2              US,PS,UIDOT,PIDOT,
     *             FIMZ,ZMDOT,
     3              TENA,FIRO,FIM,FIE,GIP,VOL,Z,DZ)
      IF (DABS(T-TMUD).LT.1.D-8) TMUD=TMUD+DTMUD
 2    CONTINUE
      DTKOD=DT
      IF (T.LT.TMAX-1.D-8) GO TO 1
      RETURN
      END
      SUBROUTINE NETUNM
      IMPLICIT REAL*8(A-H,O-Z,$)
      COMMON /AB/A(50)
      EQUIVALENCE (L,A(1))
      EQUIVALENCE (LL,A(2)),(T,A(3)),(DT,A(4)),(TMAX,A(5)),
     1            (TMUD,A(6)),(DTMUD,A(7)),(JOB,A(8)),(NERI,A(9)),
     2            (JJJ,A(10)),(KEYMON,A(11)),(NCYC,A(12))
      EQUIVALENCE (COLELA,A(13))
      EQUIVALENCE (LAGEUL,A(14))
      EQUIVALENCE (KEYEK,A(16))
      EQUIVALENCE (NCYCPR,A(17))
      EQUIVALENCE (STAB,A(18)),(DTBA,A(19)),(DTKOD,A(20)),(KDT,A(21))
      COMMON/DETO/QDET,TC,RATE,PCJDET,RCJDET,UCJDET,DCJDET,PODET,ROODET
      COMMON/DIFFUS/U2,P2,RO2,ARW
      COMMON /DRAW/GODELX,GODELY,UMIN,UMAX,PMIN,PMAX,ROMIN,ROMAX
     1             ,XMIN,XMAX,SMIN,SMAX,IVERSA
      COMMON /GAM/GAMA,NG,MU2,G1,G2,G3,G4,G5,G6,G7,G8,G9,G10,G11
     1          ,G12,G13,G14,G15,G16,G17,G18,G19,G20
      REAL*8 NG
      REAL*8 MU2
      NAMELIST /IN/LIN,GAMA,DT,TMUD,DTMUD,TMAX,
     1             GODELX,GODELY,UMIN,UMAX,PMIN,PMAX,ROMIN,ROMAX,
     2             SMIN,SMAX,IVERSA,KEYMON,COLELA,STAB
     3            ,LAGEUL,KEYEK
     4            ,QDET,TC,RATE
C*********************************************************************
      LIN=L
      LAGEUL=2
      NCYCPR=20
```

```
      KEYEK=1
      TMUD=0.
      DTMUD=0.2D0
      TMAX=2.0D0
      DT=2.5D-3
      GAMA=1.54D0
      KEYMON=1
      STAB=0.6D0
C     RATE=1.5D0
      GODELX=16.
      GODELY=20.
      IVERSA=100
      UMIN=0.
      UMAX= 3.D0
      PMIN=0.
      PMAX=1.D-3
      ROMIN=0.
      ROMAX=1.D-3
      SMIN=0.
      SMAX=PMAX/ROMAX**GAMA
      COLELA=0.
C     READ IN
C
      PRINT IN
C
      GG=2.*GAMA/(GAMA-1.)
      NG=GG
   10 CONTINUE
      MU2=(GAMA-1.)/(GAMA+1.)
      G1=GAMA-1.
      G2=1.-MU2
      G3=2./(3.*GAMA-1.)
      G4=(GAMA+1.)/2.
      G5=0.5*(3.*GAMA-1.)/(GAMA+1.)
      G6=(GAMA+1.)/(2.*GAMA)
      G7=2./(GAMA-1.)
      G8=(GAMA-1.)/(2.*GAMA)
      G9=(GAMA+1.)/(4.*GAMA)
      G10=1./GAMA
      G11=(GAMA+1.)/4.
      G12=GAMA/(GAMA-1.)
      G13=0.5*(GAMA-3.)/(GAMA+1.)
      G14=0.5*(3.*GAMA-5.)/(GAMA+1.)
      G17=GAMA+1.
      GODELX=GODELX/2.54
      GODELY=GODELY/2.54
C     CALL NAMPLT(IVERSA)
C     CALL LIMIT(1000.)
C     CALL PLOT(0.,0.5,-3)
C DATA FOR DIFFUSER ENTRY CONDITIONS. TEST III/2, REF.(M-1).
C UNITS ARE IN M,K,MILISEC.
C  FH2 IS THE DIFFUSER ENTRY MOLAR FRACTION OF H ATOMS.
      FH2=0.091D0
      U2=2.3D0
      AMDOT=1.499D-5
      A2=1.D0*7.D0*0.0254**2
      RO2=AMDOT/(A2*U2)
      TEMP0=1400.D0
      W2=7.27D0
      ARW=8.31441D-3/W2
      DELTAQ=435.783/2.D0
      QDET=FH2*DELTAQ/W2
      AA=U2/DSQRT(GAMA*ARW*TEMP0)
      RATE=1.4D0*1.D7*FH2/W2**2
      RATE=RATE*1.D2
      EM2=AA/DSQRT(1.-AA**2/G7)
      GOREM=(1.D0+EM2**2/G7)
      TRATIO=2.D0*(GAMA+1.D0)*EM2**2*GOREM/(1.D0+GAMA*EM2**2)**2
      HH0=GAMA*ARW*TEMP0/(GAMA-1.D0)
      HH1=HH0/TRATIO
      CHOKE=QDET/(HH1-HH0)
      TEMP2=TEMP0/GOREM
```

```
      P2=ARW*RO2*TEMP2
      RATIO=2.5D0/0.5D0
      EMOUT=EM2*RATIO**((GAMA-1.D0)/2.D0)
      DO 250 I=1,100
      EMOUT1=EMOUT
      GOREM=RATIO*EMOUT*(1.D0+0.5D0*(GAMA-1.D0)*EM2**2)**(0.5D0/MU2)/EM2
      GOREM=GOREM**(2.D0*MU2)
      EMOUT=DSQRT((2.D0/(GAMA-1.D0))*(GOREM-1.D0))
      DM=DABS(EMOUT-EMOUT1)
      IF(DM.LT.1.D-9) GO TO 251
  250 CONTINUE
      CALL SOF(251)
  251 CONTINUE
      PRINT 201
  201 FORMAT(//1X,'DIFFUSER ENTRY DATA')
      PRINT 202,U2,P2,RO2,FH2
  202 FORMAT(/1X,'U2,P2,RO2,FH2=',4D16.5)
      PRINT 203,AMDOT,W2,TEMP0,TEMP2,EM2
  203 FORMAT(/1X,'AMDOT,W2,TEMP0,TEMP2,EM2=',5D16.5)
      PRINT 204,RATE,QDET,HH0,HH1,CHOKE
  204 FORMAT(/1X,'RATE,QDET,HH0,HH1,CHOKE=',5D16.5/)
      PRINT 205,RATIO,EMOUT
  205 FORMAT(1X,'EXIT AREA RATIO AND MACH NUMBER  RATIO,EMOUT=',2D16.5)
      RETURN
      END
      SUBROUTINE BEGIN
     1            (L,X,U,P,RO,G,E,DU,DP,DRO,DG,DXSI,MIN,
     2             US,PS,UIDOT,PIDOT,
     *            FIMZ,ZMDOT,
     3             TENA,FIRO,FIM,FIE,GIP,VOL,Z,DZ)
      IMPLICIT REAL*8(A-H,O-Z,$)
      DIMENSION X(L),U(L),P(L),RO(L),G(L),E(L),DU(L),DP(L),DRO(L),
     1          DG(L),DXSI(L),MIN(L),
     2          US(L),PS(L),UIDOT(L),PIDOT(L)
     3         ,TENA(L),FIRO(L),FIM(L),FIE(L)
     4         ,GIP(L),VOL(L),Z(L),DZ(L)
     5         ,FIMZ(L),ZMDOT(L)
      COMMON /AB/A(50)
      COMMON /GAM/GAMA,NG,MU2,G1,G2,G3,G4,G5,G6,G7,G8,G9,G10,G11
     1         ,G12,G13,G14,G15,G16,G17,G18,G19,G20
      REAL*8 NG
      REAL*8 MU2
      EQUIVALENCE (LL,A(2))
      EQUIVALENCE (LAGEUL,A(14))
      EQUIVALENCE (UGAL,A(15))
      EQUIVALENCE (STAB,A(18)),(DTBA,A(19)),(DTKOD,A(20)),(KDT,A(21))
      COMMON/DETO/QDET,TC,RATE,PCJDET,RCJDET,UCJDET,DCJDET,PODET,ROODET
      COMMON /DRAW/GODELX,GODELY,UMIN,UMAX,PMIN,PMAX,ROMIN,ROMAX
     1            ,XMIN,XMAX,SMIN,SMAX,IVERSA
C***********************************************************************
      DTBA=0.
      DTKOD=0.
      KDT=0
      P0=1.D-7
      RHO0=1.D-7
      P1=0.1
      P1=4.
      RHO1=0.125
      RHO1=4.
      UGAL=0.
      U0=1.08D0
      U1=0.
      U0=U0-UGAL
      U1=U1-UGAL
      UMIN=UMIN-UGAL
      UMAX=UMAX-UGAL
      X0=0.625D0
      X1=3.125D0
      XMIN=X0
      XMAX=X1
      DX=(X1-X0)/(L-2.)
      DO 1 I=2,L
```

```
      X(I)=X0+(I-2.)*DX
 1    CONTINUE
      X(L)=X1
      XD=1.D10
      DO 2 I=2,LL
      U(I)=U0/X(I)
      P(I)=P0
      RO(I)=RHO0
      Z(I)=0.
      IF (X(I)-XD.LT.-1.D-8) GO TO 21
      U(I)=U1
      P(I)=P1
      RO(I)=RHO1
      Z(I)=1.D0
 21   CONTINUE
      GO TO (31,32), LAGEUL
 31   CONTINUE
      E(I)=P(I)/((GAMA-1.)*RO(I))+0.5*U(I)**2+Z(I)*QDET
      GO TO 30
 32   CONTINUE
      E(I)=P(I)/(GAMA-1.)+0.5*RO(I)*U(I)**2+Z(I)*RO(I)*QDET
 30   CONTINUE
      G(I)=DSQRT(GAMA*P(I)*RO(I))
 2    CONTINUE
      DO 3 I=2,LL
      DXSI(I)=(X(I+1)-X(I))*RO(I)
      TENA(I)=RO(I)*U(I)
      VOL(I)=(X(I+1)-X(I))*(X(I+1)+X(I))/2.D0
 3    CONTINUE
      RETURN
      END
      DOUBLE PRECISION FUNCTION RATIO(X)
      IMPLICIT REAL*8(A-H,O-Z,$)
      IF(X.LE.1.D-8)RATIO=0.
      IF(X.LE.1.D-8)RETURN
      RATIO=1.D0/X
      RETURN
      END
      DOUBLE PRECISION FUNCTION CROSS(X)
      IMPLICIT REAL*8(A-H,O-Z,$)
       CROSS=X
       RETURN
       END
      SUBROUTINE CYCEUL
     1            (L,X,U,P,RO,G,E,DU,DP,DRO,DG,DXSI,MIN,
     2             US,PS,UIDOT,PIDOT,
     *            FIMZ,ZMDOT,
     3             TENA,FIRO,FIM,FIE,GIP,VOL,Z,DZ)
      IMPLICIT REAL*8(A-H,O-Z,$)
      DIMENSION X(L),U(L),P(L),RO(L),G(L),E(L),DU(L),DP(L),DRO(L),
     1          DG(L),DXSI(L),MIN(L),
     2          US(L),PS(L),UIDOT(L),PIDOT(L)
     3         ,TENA(L),FIRO(L),FIM(L),FIE(L)
     4         ,GIP(L),VOL(L),Z(L),DZ(L)
     5         ,FIMZ(L),ZMDOT(L)
      COMMON /AB/A(50)
      EQUIVALENCE (LL,A(2)),(T,A(3)),(DT,A(4)),(COLELA,A(13))
      EQUIVALENCE (KEYEK,A(16))
      EQUIVALENCE (STAB,A(18)),(DTBA,A(19)),(DTKOD,A(20)),(KDT,A(21))
      COMMON /GAM/GAMA,NG,MU2,G1,G2,G3,G4,G5,G6,G7,G8,G9,G10,G11
     1          ,G12,G13,G14,G15,G16,G17,G18,G19,G20
      REAL*8 NG
      REAL*8 MU2
      COMMON /TOT/AMTOT,ETOT,EKTOT,EPTOT,TENTOT
      COMMON /AZOV/ISAFA,NORIMN,USAF,PSAF,ROSAF,GSAF,ESAF,DPSAF
     1            ,DXSIL,DXSIR
      LOGICAL NORIMN
      COMMON /STEP0/UL,PL,ROL,GL,UR,PR,ROR,GR,USTAR,PSTAR,
     1             RSTARL,RSTARR,GSTARL,GSTARR,WL,WR,
     2             CL,CR,CSTARL,CSTARR,UW(6)
     3            ,ZL,ZR,ZSTARL,ZSTARR,NFLUX,HELEML,HELEMR
      LOGICAL HELEML,HELEMR
```

```
      COMMON /STEP1/DUIDT,DPIDT,DGIDTL,DGIDTR,DRIDTL,DRIDTR
     2  ,ASTARL,ASTARR
     3   ,RAT,SH
      COMMON /GRADS/DUDXIL,DPDXIL,DGDXIL,DRDXIL,
     1               DUDXIR,DPDXIR,DGDXIR,DRDXIR
      COMMON /FI/FIH1,FIH2,FIH3,UXN,PXN,GXN,ROXN
     1   ,GIH
     2   ,FIH4,ZMDOTL,ZMDOTR
      COMMON/DETO/QDET,TC,RATE,PCJDET,RCJDET,UCJDET,DCJDET,P0DET,R00DET
C     DATA KOTZ/77777777777B/
C**********************************************************************
      TN=T
      DT2=DT/2.
      DO 3 I=1,L
C     MIN0=MIN(I).AND.KOTZ
C     MIN(I)=SHIFT(MIN0,30)
    3 CONTINUE
      UXN=0.
      PXN=0.
      GXN=0.
      ZN=0.
      ROXN=0.
      DO 1 I=2,L
      IM=I-1
      UXNM=UXN
      PXNM=PXN
      GXNM=GXN
      ROXNM=ROXN
      ZNM=ZN
      UL=U(IM)+0.5*DU(IM)
      PL=P(IM)+0.5*DP(IM)
      ROL=RO(IM)+0.5*DRO(IM)
      GL=G(IM)+0.5*DG(IM)
      CL=GL/ROL
      ZL=Z(IM)+0.5*DZ(IM)
      UR=U(I)-0.5*DU(I)
      PR=P(I)-0.5*DP(I)
      GR=G(I)-0.5*DG(I)
      ROR=RO(I)-0.5*DRO(I)
      CR=GR/ROR
      ZR=Z(I)-0.5*DZ(I)
      CALL RIEMAN(L,I,MIN)
      DUDXIL=DU(IM)/DXSI(IM)
      DPDXIL=DP(IM)/DXSI(IM)
      DGDXIL=DG(IM)/DXSI(IM)
      DRDXIL=DRO(IM)/DXSI(IM)
      DUDXIR=DU(I)/DXSI(I)
      DPDXIR=DP(I)/DXSI(I)
      DGDXIR=DG(I)/DXSI(I)
      DRDXIR=DRO(I)/DXSI(I)
         SH=CROSS(X(I))
         RAT=RATIO(X(I))
      CALL MAGA(L,I,MIN)
      US(I)=USTAR
      PS(I)=PSTAR
      UIDOT(I)=DUIDT
      PIDOT(I)=DPIDT
      CALL FLUXE1(L,I,MIN)
      CALL FLUXE2(L,I,MIN)
      FIRO(I)=FIH1
      FIM (I)=FIH2
      FIE (I)=FIH3
        GIP(I)=GIH
      FIMZ(I)=FIH4
      DU(IM)=UXN-UXNM
      DP(IM)=PXN-PXNM
      DG(IM)=GXN-GXNM
      DRO(IM)=ROXN-ROXNM
    1 CONTINUE
      AMTOT=0.
      ETOT=0.
      EKTOT=0.
```

```
      EPTOT=0.
      TENTOT=0.
      FI1=FIRO(2)
      FI2=FIM (2)
      FI4=FIMZ(2)
        GI2=GIP(2)
        SH=CROSS(X(2))
      DO 2 I=2,LL
      IP=I+1
      FIM1=FI1
      FIM2=FI2
      FIM4=FI4
        GIM2=GI2
        SHM=SH
       GI2=GIP(IP)
        SH=CROSS(X(IP))
        DVOL=VOL(I)
      FI1=FIRO(IP)
      FI2=FIM (IP)
      FI4=FIMZ(IP)
      ZKODM=Z(I)*RO(I)
      DX=X(IP)-X(I)
         RO(I)=RO(I)-DT/DVOL*(SH*FI1-SHM*FIM1)
         TENA(I)=TENA(I)-DT/DVOL*(SH*FI2-SHM*FIM2)-DT/DX*(GI2-GIM2)
      U(I)=TENA(I)/RO(I)
      IF(QDET.EQ.0.) GO TO 2
      Z(I)=(ZKODM-(DT/DVOL)*(SH*FI4-SHM*FIM4))/RO(I)
      DZZ1=Z(I)-1.D0
      IF(DZZ1.LT.1.D-6.AND.Z(I).GE.0.) GO TO 7350
      IF(DZZ1.LT.0.) GO TO 7352
      Z(I)=1.D0
      IF(DZZ1.LT.0.2D0) GO TO 7350
 7352 CONTINUE
      IF(Z(I).GT.0.) GO TO 7353
      Z(I)=0.
      IF(DABS(Z(I)).LT.0.2D0) GO TO 7350
 7353 CONTINUE
      PRINT 7351,I,Z(I),P(I),RO(I),E(I),U(I),ZMDOT(I)
 7351 FORMAT(/1X,'CYCEUL. I,Z,P,RO,E,U,ZMDOT=',I5,6D15.5/)
      CALL SOF(7350)
 7350 CONTINUE
      IF(Z(I).GT..1D1) Z(I)=.1D1
 2    CONTINUE
      IF(COLELA.EQ.0.) GO TO 201
      CALL DCOLE(L,X,U ,DU ,MIN,1)
      CALL DCOLE(L,X,RO,DRO,MIN,4)
 201  CONTINUE
      CALL BDOK1(L,X,U ,DU ,MIN,1)
      CALL BDOK1(L,X,RO,DRO,MIN,4)
      FI3=FIE (2)
          SH=CROSS(X(2))
      DO 6 I=2,LL
      IP=I+1
         SHM=SH
         SH=CROSS(X(IP))
         DVOL=VOL(I)
      FIM3=FI3
      FI3=FIE (IP)
      DX=X(IP)-X(I)
         E(I)=E(I)-DT/DVOL*(SH*FI3-SHM*FIM3)
C
      UAV=U(I)
C
      ROAV=RO(I)
C
      EP=E(I)-0.5*ROAV*UAV**2
      IF (QDET.EQ.0.) GO TO 64
      TI=P(I)
      DZZ=-RATE*DT*(RO(I)*Z(I))**2
      Z(I)=Z(I)+DZZ
      IF(Z(I).LT.0.) Z(I)=0.
      IF (Z(I).LT.0.01) Z(I)=0.
```

```
      EP=EP-Z(I)*RO(I)*QDET
 64   CONTINUE
      GO TO (61,62), KEYEK
 61   CONTINUE
      GO TO 60
 62   CONTINUE
      DEK=(ROAV*DU(I)**2+2.*DRO(I)*DU(I)*UAV)/24.
      EP=EP-DEK
      GO TO 60
 60   CONTINUE
      IF(EP.LE.0.) GO TO 7001
      P(I)=G1*EP
      G(I)=DSQRT(GAMA*P(I)*RO(I))
      DM=RO(I)*DX
      DXSI(I)=DM
         DM=RO(I)*DVOL
         ETOT=ETOT+E(I)*DVOL
         EPTOT=EPTOT+EP*DVOL
      AMTOT=AMTOT+DM
      ETOT=ETOT+E(I)*DX
      EPTOT=EPTOT+EP*DX
      TENTOT=TENTOT+DX*TENA(I)
      UPC=DABS(U(I))+G(I)/RO(I)
      DTI=STAB*DX/UPC
      IF(DTI.GT.DTBA) GO TO 69
      DTBA=DTI
      KDT=I
 69   CONTINUE
 6    CONTINUE
      CALL DCOLE(L,X,Z,DZ,MIN,5)
      IF(COLELA.EQ.0.) GO TO 601
      CALL DCOLE(L,X,P,DP,MIN,2)
      CALL DCOLE(L,X,G,DG,MIN,3)
 601  CONTINUE
      CALL BDOK1(L,X,P,DP,MIN,2)
      CALL BDOK1(L,X,G,DG,MIN,3)
      CALL BDOK1(L,X,Z,DZ,MIN,5)
      EKTOT=ETOT-EPTOT
      RETURN
 7001 CONTINUE
      PRINT 7101, I,ROAV,UAV,DRO(I),DU(I),E(I),DEK,EP,KEYEK
 7101 FORMAT(//1X,'FROM CYCEUL.  NEGATIVE EP.  IN CELL  I=',I6//
     1          1X,'ROAV,UAV,DRO(I),DU(I)=',4D18.8//
     2          1X,'E(I),DEK,EP,KEYEK=',3D18.8,I10//)
      CALL SOF(7001)
      RETURN
      END
      SUBROUTINE SAFAE
     1             (L,X,U,P,RO,G,E,DU,DP,DRO,DG,DXSI,MIN,
     2              US,PS,UIDOT,PIDOT,
     *             FIMZ,ZMDOT,
     3              TENA,FIRO,FIM,FIE,GIP,VOL,Z,DZ)
      IMPLICIT REAL*8(A-H,O-Z,$)
      DIMENSION X(L),U(L),P(L),RO(L),G(L),E(L),DU(L),DP(L),DRO(L),
     1          DG(L),DXSI(L),MIN(L),
     2          US(L),PS(L),UIDOT(L),PIDOT(L)
     3         ,TENA(L),FIRO(L),FIM(L),FIE(L)
     4         ,GIP(L),VOL(L),Z(L),DZ(L)
     5         ,FIMZ(L),ZMDOT(L)
      COMMON /AB/A(50)
      EQUIVALENCE (LL,A(2)),(T,A(3)),(DT,A(4)),(NCYC,A(12))
      COMMON /GAM/GAMA,NG,MU2,G1,G2,G3,G4,G5,G6,G7,G8,G9,G10,G11
     1          ,G12,G13,G14,G15,G16,G17,G18,G19,G20
      REAL*8 NG
      REAL*8 MU2
      COMMON/DETO/QDET,TC,RATE,PCJDET,RCJDET,UCJDET,DCJDET,PODET,ROODET
      COMMON/DIFFUS/U2,P2,RO2,ARW
C**********************************************************************
      TN=T
C
C  INFLOW B.C. AT I=2
C
```

```
      U(1)=U2
      P(1)=P2
      RO(1)=RO2
      G(1)=DSQRT(GAMA*P(1)*RO(1))
      Z(1)=1.D0
      DU(1)=0.
      DP(1)=0.
      DG(1)=0.
      DRO(1)=0.
      DXSI(1)=DXSI(2)
C
C  OUTFLOW B.C. AT I=L
C
      U(L)=U(LL)
      P(L)=P(LL)
      RO(L)=RO(LL)
      G(L)=DSQRT(GAMA*P(L)*RO(L))
      DU(L)=0.
      DP(L)=0.
      DG(L)=0.
      DRO(L)=0.
      DXSI(L)=DXSI(LL)
C
      RETURN
      END
      SUBROUTINE BDOK1(L,X,V,DV,MIN,NV)
      IMPLICIT REAL*8(A-H,O-Z,$)
      DIMENSION X(L),V(L),DV(L),MIN(L)
      COMMON /AB/A(50)
      EQUIVALENCE (LL,A(2)),(KEYMON,A(11))
      COMMON /DRAW/GODELX,GODELY,UMIN,UMAX,PMIN,PMAX,ROMIN,ROMAX
     1          ,XMIN,XMAX,SMIN,SMAX,IVERSA
      COMMON /MONIT/NC14(4),CASEAV(4),NF16(6),
     1                NMONU(4),NMONP(4),NMONG(4),NMONRO(4),NMONZ(4)
      DIMENSION NMONV(4,5)
      EQUIVALENCE (NMONV(1,1),NMONU(1))
C     DIMENSION NAMEV(5)
C     DATA NAMEV/1HU,1HP,1HG,2HRO,1HZ/
      DATA EPS/1.D-15/
C***********************************************************************
C     NV=0
C     DO 10 N=1,5
C     IF (NAME.EQ.NAMEV(N))NV=N
C10   CONTINUE
      GO TO (1,2,3,4,5), NV
 1    AMIDA=(UMAX-UMIN)**2
      GO TO 9
 2    AMIDA=(PMAX-PMIN)**2
      GO TO 9
 3    AMIDA=((UMAX-UMIN)*(ROMAX-ROMIN))**2
      GO TO 9
 4    AMIDA=(ROMAX-ROMIN)**2
      GO TO 9
 5    AMIDA=1.D0
      GO TO 9
 9    CONTINUE
      AMIDA=AMIDA*EPS**2
      AMIDA=EPS
      DO 29 I=2,LL
      ICAT=0
      VLEFT=V(I)-0.5D0*DV(I)
      VRIGHT=V(I)+0.5D0*DV(I)
      VM=V(I-1)
      VP=V(I+1)
      SIGN=(VP-V(I))*(V(I)-VM)
      IF(SIGN.GT.-AMIDA) GO TO 22
  21  DV(I)=0.
      ICAT=1
      GO TO 20
  22  CONTINUE
      SIGN=(VP-VM)*DV(I)
      IF(SIGN.GT.-AMIDA) GO TO 24
```

```
   23  DV(I)=0.5D0*(VP-VM)
       VLEFT=V(I)-0.5D0*DV(I)
       VRIGHT=V(I)+0.5D0*DV(I)
       ICAT=2
   24  SIGN=(VLEFT-VM)*DV(I)
       IF(SIGN.GT.-AMIDA) GO TO 26
   25  VLEFT=VM
       VRIGHT=2.D0*V(I)-VLEFT
       DV(I)=VRIGHT-VLEFT
       ICAT=3
   26  SIGN=(VP-VRIGHT)*DV(I)
       IF (SIGN.GT.-AMIDA) GO TO 28
   27  VRIGHT=VP
       VLEFT=2.D0*V(I)-VRIGHT
       DV(I)=VRIGHT-VLEFT
       ICAT=3
  28   IF(DABS(DV(I)).LE.0.5D0*DABS(VP-VM)) GO TO 31
  30   DV(I)=0.5D0*(VP-VM)
       ICAT=4
  31   CONTINUE
  20   CONTINUE
       IF (DV(I)**2.GT.AMIDA) GO TO 40
       DV(I)=0.
  40   CONTINUE
C      IF (ICAT.GT.0) NMONV(ICAT,NV)=NMONV(ICAT,NV)+1
C      IBYTE0=5
C      MIN(I)=MIN(I).OR.SHIFT(ICAT,(NV+IBYTE0-2)*3)
  29   CONTINUE
       RETURN
       END
       SUBROUTINE DCOLE(L,X,V,DV,MIN,NV)
       IMPLICIT REAL*8(A-H,O-Z,$)
       DIMENSION X(L),V(L),DV(L),MIN(L)
       COMMON /AB/A(50)
       EQUIVALENCE (LL,A(2))
C*****************************************************************
       DO 1 I=2,LL
       IM=I-1
       IP=I+1
       DV(I)=0.5*(V(IP)-V(IM))
  1    CONTINUE
       RETURN
       END
       SUBROUTINE PRINT
     1             (L,X,U,P,RO,G,E,DU,DP,DRO,DG,DXSI,MIN,
     2              US,PS,UIDOT,PIDOT,
     *             FIMZ,ZMDOT,
     3               TENA,FIRO,FIM,FIE,GIP,VOL,Z,DZ)
       IMPLICIT REAL*8(A-H,O-Z,$)
       DIMENSION X(L),U(L),P(L),RO(L),G(L),E(L),DU(L),DP(L),DRO(L),
     1           DG(L),DXSI(L),MIN(L),
     2           US(L),PS(L),UIDOT(L),PIDOT(L)
     3          ,TENA(L),FIRO(L),FIM(L),FIE(L)
     4          ,GIP(L),VOL(L),Z(L),DZ(L)
     5          ,FIMZ(L),ZMDOT(L)
       COMMON /TOT/AMTOT,ETOT,EKTOT,EPTOT,TENTOT
       COMMON /STEP0/UL,PL,ROL,GL,UR,PR,ROR,GR,USTAR,PSTAR,
     1               RSTARL,RSTARR,GSTARL,GSTARR,WL,WR,
     2               CL,CR,CSTARL,CSTARR,UW(6)
     3              ,ZL,ZR,ZSTARL,ZSTARR,NFLUX,HELEML,HELEMR
       LOGICAL HELEML,HELEMR
       COMMON /AB/A(50)
       EQUIVALENCE (LL,A(2)),(T,A(3)),(NCYC,A(12)),(DT,A(4))
       EQUIVALENCE (UGAL,A(15))
       COMMON/DETO/QDET,TC,RATE,PCJDET,RCJDET,UCJDET,DCJDET,P0DET,RO0DET
       COMMON/DIFFUS/U2,P2,RO2,ARW
       COMMON /GAM/GAMA,NG,MU2,G1,G2,G3,G4,G5,G6,G7,G8,G9,G10,G11
     1          ,G12,G13,G14,G15,G16,G17,G18,G19,G20
       REAL*8 NG
       REAL*8 MU2
       COMMON /MONIT/NC14(4),CASEAV(4),NF16(6),
     1                NMONU(4),NMONP(4),NMONG(4),NMONRO(4),NMONZ(4)
```

```
      DIMENSION CASAV1(4)
      LOGICAL FULLPR
      COMMON /PRNT/ROX,PX,UX,ZX
C***********************************************************************
      FULLPR=.TRUE.
      PRINT 1
1     FORMAT(1H1)
      PRINT 2, T,DT,NCYC
2     FORMAT(1X,10X,'RESULTS AT  T=',D11.5,5X,'DT=',D11.5,5X,'NCYC=',
     1            I5//)
      PRINT 3, AMTOT,ETOT,EKTOT,EPTOT,TENTOT
3     FORMAT(1X,'AMTOT=',D20.14,2X,'ETOT,EKTOT,EPTOT=',3D22.14/
     1       1X,'TENTOT=',D21.14//)
4     FORMAT(1X,'  I','     X        ','       U        ','      P       ',
     1            '     RO       ','       G        ','      Z       ',
     2            '     DU      ','     DP      ','   DRO      ',
     3            '     DG      ','     DZ')
44    FORMAT(1X,'   ','            ','       US       ','     PS       ',
     1            '   ZMDOT      ','     FIMZ      ','  AMDOT       ',
     2            '   AMDOTN    ','    TEMP     ','  ENTALP    ',
     3            '   AMACH     ','  ENTRO     ')
5     FORMAT(1X)
      IF (UGAL.NE.0.) PRINT 6, UGAL
6     FORMAT(/11X,'INITIAL VELOCITY CORRESPONDS TO  UGAL=',D15.6/)
      DO 10 I=1,L
      IF (MOD(I,10).NE.1) GO TO 11
      PRINT 5
      PRINT 4
      PRINT 44
      PRINT 5
11    CONTINUE
      PRINT 12,I,X(I),U(I),P(I),RO(I),G(I),Z(I),DU(I),DP(I),DRO(I),
     1            DG(I),DZ(I)
12    FORMAT(1X,I3,6D12.5,5D11.4)
      ENTRO=P(I)/RO(I)**GAMA
      IF(.NOT.FULLPR) GO TO 131
      IF(I.EQ.1) GO TO 131
      IM=I-1
      UL=U(IM)+0.5*DU(IM)
      PL=P(IM)+0.5*DP(IM)
      ROL=RO(IM)+0.5*DRO(IM)
      GL=G(IM)+0.5*DG(IM)
      CL=GL/ROL
      ZL=Z(IM)+0.5*DZ(IM)
      UR=U(I)-0.5*DU(I)
      PR=P(I)-0.5*DP(I)
      GR=G(I)-0.5*DG(I)
      ROR=RO(I)-0.5*DRO(I)
      CR=GR/ROR
      ZR=Z(I)-0.5*DZ(I)
      CALL RIEMAN(L,I,MIN)
      CALL FLUXE1(L,I,MIN)
      XI=X(I)
      IF(I.EQ.L) GO TO 222
      UX=U(I)
      PX=P(I)
      ROX=RO(I)
      ZX=Z(I)
      IP=I+1
      XI=0.5D0*(X(I)+X(IP))
222   CONTINUE
      AMACH=UX/DSQRT(GAMA*PX/ROX)
      AMDOT=ROX*UX*CROSS(XI)
      IF(I.EQ.2)AMDOT0=AMDOT
      AMDOTN=AMDOT/AMDOT0
      ENTALP=(GAMA/(GAMA-1.D0))*PX/ROX+0.5D0*UX**2
      TEMP=PX/(ROX*ARW)
      PRINT 13,US(I),PS(I),
     1          ZMDOT(I),FIMZ(I),AMDOT,AMDOTN,TEMP,ENTALP,AMACH,ENTRO
13    FORMAT(4X,12X,5D12.5,6D11.4)
131   CONTINUE
10    CONTINUE
```

```
C  JOB STATISTICS
C      DO 40 I=1,4
C      CASAV1(I)=0.
C      IF (NC14(I).NE.0) CASAV1(I)=CASEAV(I)/NC14(I)
C40    CONTINUE
C      PRINT 30
C30    FORMAT(///1X,10(1H*),3X,'JOB STATISTICS',3X,10(1H*)//)
C      PRINT 31,(NC14(I),I=1,4)
C31    FORMAT(1X,'NO. OF VARIOUS CASES IN RIEMAN SOLVER   NC14(NCASE)=',
C    1                  4I10)
C      PRINT 301, (CASAV1(I),I=1,4)
C301   FORMAT(/1X,'AVERAGE NUMBER OF ITERATIONS IN RIEMAN SOLVER',
C    1           1X,'  CASAV1(NCASE)=',4(F6.2,4X))
C      PRINT 32,(NF16(I),I=1,6)
C32    FORMAT(/1X,'NO. OF VARIOUS FLUX CASES   NF16(NFLUX)=',6I10)
C      ICAT0=4
C      PRINT 33,(NMONU(I),I=1,ICAT0),(NMONP(I),I=1,ICAT0),
C    1            (NMONG(I),I=1,ICAT0),(NMONRO(I),I=1,ICAT0)
C33    FORMAT(/1X,'NO. OF MONOTONICITY INTERVENTIONS FOR EACH VAR.',
C    1           1X,'IN EACH CATEGORY.'/
C    1           1X,'NMONU (ICAT)=',4I10/
C    1           1X,'NMONP (ICAT)=',4I10/
C    1           1X,'NMONG (ICAT)=',4I10/
C    1           1X,'NMONRO(ICAT)=',4I10/)
       RETURN
       END
       SUBROUTINE SOF(ISTOP)
       IMPLICIT REAL*8(A-H,O-Z,$)
       PRINT 1, ISTOP
 1     FORMAT(1X,3H***,3X,'SOF',I6,3X,3H***/)
       XX=-1.D0
       YY=DSQRT(XX)
C      CALL SYSTEM(51,16H SUBROUTINE TREE)
       STOP
       END
       SUBROUTINE RIEMAN(L,I,MIN)
       IMPLICIT REAL*8(A-H,O-Z,$)
       DIMENSION MIN(L)
       COMMON /STEP0/UL,PL,ROL,GL,UR,PR,ROR,GR,USTAR,PSTAR,
     1              RSTARL,RSTARR,GSTARL,GSTARR,WL,WR,
     2              CL,CR,CSTARL,CSTARR,UW(6)
     3             ,ZL,ZR,ZSTARL,ZSTARR,NFLUX,HELEML,HELEMR
       LOGICAL HELEML,HELEMR
       COMMON /STEP1/DUIDT,DPIDT,DGIDTL,DGIDTR,DRIDTL,DRIDTR
     2  ,ASTARL,ASTARR
     3    ,RAT,SH
       COMMON /DRAW/GODELX,GODELY,UMIN,UMAX,PMIN,PMAX,ROMIN,ROMAX
     1             ,XMIN,XMAX,SMIN,SMAX,IVERSA
       COMMON /GAM/GAMA,NG,MU2,G1,G2,G3,G4,G5,G6,G7,G8,G9,G10,G11
     1          ,G12,G13,G14,G15,G16,G17,G18,G19,G20
       REAL*8 NG
       REAL*8 MU2
       COMMON /AB/A(50)
       COMMON /MONIT/NC14(4),CASEAV(4),NF16(6),
     1               NMONU(4),NMONP(4),NMONG(4),NMONRO(4),NMONZ(4)
C***********************************************************************
       DATA NMAX/63/
       DATA EPS/1.D-8/
C***********************************************************************
       IF(PL.GT.0..AND.PR.GT.0.) GO TO 7201
       PRINT 7211,I,PL,PR
7211   FORMAT(' FROM RIEMAN. I,PL,PR=',I6,2D16.6/)
       CALL SOF(7654)
 7201  CONTINUE
       UW(6)=1.D20
       WL=0.
       WR=0.
       ZETAL=PL**G8
       ZETAR=PR**G8
       CLG=CL/GAMA
       CRG=CR/GAMA
       ZSTARL=ZL
```

```
      ZSTARR=ZR
      IF (ZETAL.LT.ZETAR) GO TO 102
C  LEFT PRESSURE IS HIGHER
 101  CONTINUE
      EVERR=(PL-PR)/PR
      USR=UR+CRG*EVERR/DSQRT(1.+G6*EVERR)
      SR=USR
      UEL=UL-G7*CL*(ZETAR-ZETAL)/ZETAL
      SL=UEL
      NL=2
      NR=2
      IF (USR.GE.UL) NL=1
      IF (UEL.LE.UR) NR=1
      IF (DABS(EVERR).LT.EPS) GO TO 100
      IF (NL.EQ.2.AND.NR.EQ.1) GO TO 7001
      GO TO 100
C  RIGHT PRESSURE IS HIGHER
 102  CONTINUE
      EVERL=(PR-PL)/PL
      USL=UL-CLG*EVERL/DSQRT(1.+G6*EVERL)
      SL=USL
      UER=UR+G7*CR*(ZETAL-ZETAR)/ZETAR
      SR=UER
      NL=2
      NR=2
      IF (UER.GE.UL) NL=1
      IF (USL.LE.UR) NR=1
      IF (DABS(EVERL).LT.EPS) GO TO 100
      IF (NL.EQ.1.AND.NR.EQ.2) GO TO 7001
      GO TO 100
 100  CONTINUE
      IF (NL.EQ.1.AND.NR.EQ.2) NCASE=1
      IF (NL.EQ.2.AND.NR.EQ.2) NCASE=2
      IF (NL.EQ.2.AND.NR.EQ.1) NCASE=3
      IF (NL.EQ.1.AND.NR.EQ.1) NCASE=4
      IF(DABS(PL-PR)+DABS(UL-UR).LT.EPS*(PMAX-UMIN)) NCASE=4
      UMIDA=EPS*DMAX1(CL,CR)
      DUDZL=-G7*CL/ZETAL
      DUDZR= G7*CR/ZETAR
      ZETA=(-(UR-UL)+ZETAR*DUDZR-ZETAL*DUDZL)/(DUDZR-DUDZL)
      IF (ZETA.LE.0.) GO TO 7002
      N=0
      GO TO (1,2,3,4), NCASE
C  THE CASE  ES
 1    ITYPE=NCASE
      HELEML=.FALSE.
      HELEMR=.TRUE.
 11   N=N+1
      IF (N.GT.NMAX) GO TO 7003
      ZETAF=ZETA
      UEL=UL-G7*CL*(ZETAF-ZETAL)/ZETAL
      PPR=(ZETAF/ZETAR)**NG
      EVERR=PPR-1.
      SQRR=DSQRT(1.+G6*EVERR)
      USR=UR+CRG*EVERR/SQRR
      DU=UEL-USR
      IF (DABS(DU).LE.UMIDA) GO TO 10
      DUDZR=NG*CRG*(PPR/ZETAF)*(1.+G9*EVERR)/SQRR**3
      ZETA=ZETAF+DU/(DUDZR-DUDZL)
      GO TO 11
 10   CONTINUE
      USTAR=(UEL+USR)/2.
      PSTAR=PPR*PR
      CSTARL=CL+(UL-USTAR)/G7
      RSTARL=GAMA*PSTAR/CSTARL**2
      GSTARL=CSTARL*RSTARL
C  EQU. NO.  69.01 OF THE BOOK BY COURANT-FRIEDRICHS.
      WWR=G11*(USTAR-UR)*ROR
      WR=WWR+DSQRT(GR**2+WWR**2)
      RSTARR=ROR*WR/(WR-ROR*(USTAR-UR))
      GSTARR=DSQRT(GAMA*PSTAR*RSTARR)
      CSTARR=GSTARR/RSTARR
```

```
      WRE=WR/ROR+UR
      UW(1)=UL-CL
      UW(2)=USTAR-CSTARL
      UW(3)=USTAR
      UW(4)=WRE
      UW(5)=WRE
      GO TO 5
C  THE CASE  SS
 2    ITYPE=NCASE
      HELEML=.TRUE.
      HELEMR=.TRUE.
 21   N=N+1
      IF (N.GT.NMAX) GO TO 7003
      ZETAF=ZETA
      PF=ZETAF**NG
      PPL=PF/PL
      PPR=PF/PR
      EVERL=PPL-1.
      EVERR=PPR-1.
      SQRL=DSQRT(1.+G6*EVERL)
      SQRR=DSQRT(1.+G6*EVERR)
      USL=UL-CLG*EVERL/SQRL
      USR=UR+CRG*EVERR/SQRR
      DU=USL-USR
      IF (DABS(DU).LE.UMIDA) GO TO 20
      DUDZL=-NG*CLG*(PPL/ZETAF)*(1.+G9*EVERL)/SQRL**3
      DUDZR= NG*CRG*(PPR/ZETAF)*(1.+G9*EVERR)/SQRR**3
      ZETA=ZETAF+DU/(DUDZR-DUDZL)
      GO TO 21
 20   CONTINUE
      USTAR=(USL+USR)/2.
      PSTAR=(PPL*PL+PPR*PR)/2.
      WWR=G11*(USTAR-UR)*ROR
      WR=WWR+DSQRT(GR**2+WWR**2)
      WWL=-G11*(USTAR-UL)*ROL
      WL=WWL+DSQRT(GL**2+WWL**2)
      RSTARL=ROL*WL/(WL+ROL*(USTAR-UL))
      RSTARR=ROR*WR/(WR-ROR*(USTAR-UR))
      GSTARL=DSQRT(GAMA*PSTAR*RSTARL)
      GSTARR=DSQRT(GAMA*PSTAR*RSTARR)
      CSTARL=GSTARL/RSTARL
      CSTARR=GSTARR/RSTARR
      WLE=-WL/ROL+UL
      WRE=WR/ROR+UR
      UW(1)=WLE
      UW(2)=WLE
      UW(3)=USTAR
      UW(4)=WRE
      UW(5)=WRE
      GO TO 5
C  THE CASE  SE
 3    ITYPE=NCASE
      HELEML=.TRUE.
      HELEMR=.FALSE.
 31   N=N+1
      IF (N.GT.NMAX) GO TO 7003
      ZETAF=ZETA
      UER=UR+G7*CR*(ZETAF-ZETAR)/ZETAR
      PPL=(ZETAF/ZETAL)**NG
      EVERL=PPL-1.
      SQRL=DSQRT(1.+G6*EVERL)
      USL=UL-CLG*EVERL/SQRL
      DU=USL-UER
      IF (DABS(DU).LE.UMIDA) GO TO 30
      DUDZL=-NG*CLG*(PPL/ZETAF)*(1.+G9*EVERL)/SQRL**3
      ZETA=ZETAF+DU/(DUDZR-DUDZL)
      GO TO 31
 30   CONTINUE
      USTAR=(USL+UER)/2.
      PSTAR=PPL*PL
      CSTARR=CR-(UR-USTAR)/G7
      RSTARR=GAMA*PSTAR/CSTARR**2
```

```
      GSTARR=CSTARR*RSTARR
      WWL=-G11*(USTAR-UL)*ROL
      WL=WWL+DSQRT(GL**2+WWL**2)
      WLE=-WL/ROL+UL
      RSTARL=ROL*WL/(WL+ROL*(USTAR-UL))
      GSTARL=DSQRT(GAMA*PSTAR*RSTARL)
      CSTARL=GSTARL/RSTARL
      UW(1)=WLE
      UW(2)=WLE
      UW(3)=USTAR
      UW(4)=USTAR+CSTARR
      UW(5)=UR+CR
      GO TO 5
C  THE CASE  EE
 4    ITYPE=NCASE
      HELEML=.FALSE.
      HELEMR=.FALSE.
      PSTAR=ZETA**NG
      USTAR=UL-G7*CL*(ZETA-ZETAL)/ZETAL
      CSTARL=CL+(UL-USTAR)/G7
      CSTARR=CR-(UR-USTAR)/G7
      RSTARL=GAMA*PSTAR/CSTARL**2
      RSTARR=GAMA*PSTAR/CSTARR**2
      GSTARL=RSTARL*CSTARL
      GSTARR=RSTARR*CSTARR
      UW(1)=UL-CL
      UW(2)=USTAR-CSTARL
      UW(3)=USTAR
      UW(4)=USTAR+CSTARR
      UW(5)=UR+CR
      N=1
      GO TO 5
 5    CONTINUE
      DO 6 K=1,6
      NFLUX=K
      IF (UW(K).GE.0.) GO TO 61
 6    CONTINUE
      NFLUX=6
 61   CONTINUE
C     MIN(I)=MIN(I).OR.NCASE
C     MIN(I)=MIN(I).OR.SHIFT(N,3)
C     MIN(I)=MIN(I).OR.SHIFT(NFLUX,9)
C     NC14(NCASE)=NC14(NCASE)+1
C     CASEAV(NCASE)=CASEAV(NCASE)+N
C     NF16(NFLUX)=NF16(NFLUX)+1
      IF(A(3).NE.0.)GO TO 666
      IF(I.NE.2) GO TO 666
      PRINT 667,I,NFLUX,NCASE,PL,UL,ROL,PR,UR,ROR,USTAR,PSTAR,RSTARL,
     1          RSTARR,(KK,UW(KK),KK=1,6)
667   FORMAT(/1X,'I,NFLUX,NCASE=',3I5/1X,'PL,UL,ROL,PR,UR,ROR=',6D12.4/
     1         1X,'USTAR,PSTAR,RSTARL,RSTARR=',4D13.4/
     2         1X,'KK,UW(KK)=',6(I4,2X,D13.4)/)
 666  CONTINUE
      RETURN
 7001 CONTINUE
      PRINT 7101, PL,UL,PR,UR,ZETAL,ZETAR,SL,SR,NL,NR,I
 7101 FORMAT(//1X,'FROM RIEMAN.  AN IMPOSSIBLE CASE OF EXPANSION/SHOCK'
     1       //1X,'PL,UL,PR,UR=',4D25.14//
     2          1X,'ZETAL,ZETAR,SL,SR=',4D25.14//
     3          1X,'NL,NR,I=',3I10//)
      CALL SOF(7001)
 7002 CONTINUE
      PRINT 7102, ZETA,DUDZL,DUDZR,ZETAL,ZETAR,PL,UL,PR,UR,N,NCASE,I
 7102 FORMAT(//1X,'FROM RIEMAN.  NEGATIVE PRESSURE AT THE INTERSECTION',
     1          1X,'OF L AND R EXPANSION BRANCHES'//
     2          1X,'IT MEANS THAT A CAVITATION TENDS TO FORM.  THIS',
     3          1X,'POSSIBILITY IS EXCLUDED IN PRESENT VERSION'//
     4          1X,'ZETA,DUDZL,DUDZR,ZETAL,ZETAR,PL,UL,PR,UR=',9D10.3//
     5          1X,'N,NCASE,I=',3I10//)
      CALL SOF(7002)
 7003 CONTINUE
      PRINT 7103, I,N,NCASE,DU,UMIDA,EPS,PL,UL,PR,UR,
```

```
     1              ZETA,ZETAF,ZETAL,ZETAR,DUDZL,DUDZR
7103 FORMAT(//1X,'FROM RIEMAN.  NUMBER OF ITERATIONS EXCEEDED.'//
     1          1X,'I,N,NCASE,DU,UMIDA,EPS=',3I6,3D18.6//
     2          1X,'PL,UL,PR,UR,ZETA,ZETAF=',6D18.10//
     3          1X,'ZETAL,ZETAR,DUDZL,DUDZR=',4D18.10//)
      CALL SOF(7003)
      RETURN
      END
      SUBROUTINE MAGA(L,I,MIN)
      IMPLICIT REAL*8(A-H,O-Z,$)
      DIMENSION MIN(L)
      COMMON /GAM/GAMA,NG,MU2,G1,G2,G3,G4,G5,G6,G7,G8,G9,G10,G11
     1       ,G12,G13,G14,G15,G16,G17,G18,G19,G20
      REAL*8 NG
      REAL*8 MU2
      COMMON /STEP0/UL,PL,ROL,GL,UR,PR,ROR,GR,USTAR,PSTAR,
     1              RSTARL,RSTARR,GSTARL,GSTARR,WL,WR,
     2              CL,CR,CSTARL,CSTARR,UW(6)
     3              ,ZL,ZR,ZSTARL,ZSTARR,NFLUX,HELEML,HELEMR
      LOGICAL HELEML,HELEMR
      COMMON /STEP1/DUIDT,DPIDT,DGIDTL,DGIDTR,DRIDTL,DRIDTR
     2  ,ASTARL,ASTARR
     3   ,RAT,SH
      COMMON /GRADS/DUDXIL,DPDXIL,DGDXIL,DRDXIL,
     1              DUDXIR,DPDXIR,DGDXIR,DRDXIR
      COMMON /AB/A(50)
      REAL LU,LP,LRO
      DATA EPS/1.D-6/
C*********************************************************************
C  WE HERE SOLVE FOR THE TIME-DERIVATIVES ALONG THE CONTACT SURFACE,
C  NAMELY  DUIDT,DPIDT.  FROM THESE WE ALSO OBTAIN THE OTHER
C  TIME-DERIVATIVES (SEE COMMON /STEP1/).
C  WE COMPUTE THE COEFFICIENTS FOR TWO EQUATIONS FOR DUIDT,DPIDT.  THESE
C  ARE          AAL*DUIDT+BBL*DPIDT=CCL
C               AAR*DUIDT+BBR*DPIDT=CCR
C*********************************************************************
C     IF(SH.LE.EPS)RAT=0.
      IF (.NOT.HELEML) GO TO 12
 11   CONTINUE
      DP=PSTAR-PL
      IF (DABS(DP).LT.EPS*PSTAR) GO TO 12
      DU=USTAR-UL
      Z1=1./DP
      Z2=0.5/(PSTAR+MU2*PL)
      ZA=Z1-Z2
      ZB=Z1+MU2*Z2
      LU=-ROL*DU*(GAMA*PL*ZB+0.5)+WL
      LRO=0.5*DP/ROL
      LP=1.+ZB*DP
      AAL=1.+ZA*DP
      BBL=-ZA*DU+WL/GSTARL**2
      CCL=-LU*DUDXIL-LRO*DRDXIL-LP*DPDXIL
      CCL=CCL-WL*USTAR*RAT/RSTARL
     1  +UL*RAT*DU*(GAMA*PL*ZB+0.5)
      GO TO 10
 12   CONTINUE
      A1=DUDXIL+DPDXIL/GL
      BETA=GSTARL/GL
      ASTARL=A1+(G3/GL)*(CL*DGDXIL-G4*DPDXIL)*(BETA**G5-1.)
      AAL=1.
      BBL=1./GSTARL
      CCL=-GSTARL*ASTARL/DSQRT(BETA)
      GEOM=RAT*((GAMA-1.)*UL+2.*CL)*
     1 (BETA**G13-1.)/(ROL*(GAMA-3.))
     1  -4.*RAT*CL*(BETA**G14-1.)/(ROL*(3.*GAMA-5.))
      ASTARL=ASTARL-GEOM
      EVER1= GSTARL*GEOM/DSQRT(BETA)
      EVER2=-RAT*USTAR*CSTARL
      CCL=CCL+EVER1+EVER2
      GO TO 10
 10   CONTINUE
      IF (.NOT.HELEMR) GO TO 22
```

```
 21     CONTINUE
        DP=PSTAR-PR
        IF (DABS(DP).LT.EPS*PSTAR) GO TO 22
        DU=USTAR-UR
        Z1=1./DP
        Z2=0.5/(PSTAR+MU2*PR)
        ZA=Z1-Z2
        ZB=Z1+MU2*Z2
        LU=-ROR*DU*(GAMA*PR*ZB+0.5)-WR
        LRO=0.5*DP/ROR
        LP=1.+ZB*DP
        AAR=1.+ZA*DP
        BBR=-ZA*DU-WR/GSTARR**2
        CCR=-LU*DUDXIR-LRO*DRDXIR-LP*DPDXIR
        CCR=CCR+WR*USTAR*RAT/RSTARR
       1  +UR*RAT*DU*(GAMA*PR*ZB+0.5)
        GO TO 20
 22     CONTINUE
        A1=DUDXIR-DPDXIR/GR
        BETA=GSTARR/GR
        ASTARR=A1-(G3/GR)*(CR*DGDXIR-G4*DPDXIR)*(BETA**G5-1.)
        AAR=1.
        BBR=-1./GSTARR
        CCR=GSTARR*ASTARR/DSQRT(BETA)
        GEOM=RAT*(-(GAMA-1.)*UR+2.*CR)*(BETA**G13-1.)
       1  /(ROR*(GAMA-3.))
       2  -4.*RAT*CR*(BETA**G14-1.)/(ROR*(3.*GAMA-5.))
        ASTARR=ASTARR+GEOM
        EVER1=GSTARR*GEOM/DSQRT(BETA)
         EVER2=RAT*USTAR*CSTARR
        CCR=CCR+EVER1+EVER2
        GO TO 20
 20     CONTINUE
        DET=AAL*BBR-AAR*BBL
        DUIDT=(CCL*BBR-CCR*BBL)/DET
        DPIDT=-(CCL*AAR-CCR*AAL)/DET
        DRIDTL=DPIDT/CSTARL**2
        DRIDTR=DPIDT/CSTARR**2
        DGIDTL=0.5*GSTARL*(DPIDT/PSTAR+DRIDTL/RSTARL)
        DGIDTR=0.5*GSTARR*(DPIDT/PSTAR+DRIDTR/RSTARR)
        RETURN
        END
```

---

```
        SUBROUTINE FLUXE(L,I,MIN)
        IMPLICIT REAL*8(A-H,O-Z,$)
        DIMENSION MIN(L)
        COMMON /AB/A(50)
        EQUIVALENCE (DT,A(4)),(NCYC,A(12))
        COMMON /GAM/GAMA,NG,MU2,G1,G2,G3,G4,G5,G6,G7,G8,G9,G10,G11
       1          ,G12,G13,G14,G15,G16,G17,G18,G19,G20
        REAL*8 NG
        REAL*8 MU2
        COMMON /GRADS/DUDXIL,DPDXIL,DGDXIL,DRDXIL,
       1              DUDXIR,DPDXIR,DGDXIR,DRDXIR
        COMMON /STEP0/UL,PL,ROL,GL,UR,PR,ROR,GR,USTAR,PSTAR,
       1              RSTARL,RSTARR,GSTARL,GSTARR,WL,WR,
       2              CL,CR,CSTARL,CSTARR,UW(6)
       3             ,ZL,ZR,ZSTARL,ZSTARR,NFLUX,HELEML,HELEMR
        LOGICAL HELEML,HELEMR
        COMMON /STEP1/DUIDT,DPIDT,DGIDTL,DGIDTR,DRIDTL,DRIDTR
       2  ,ASTARL,ASTARR
       3   ,RAT,SH
        COMMON/DETO/QDET,TC,RATE,PCJDET,RCJDET,UCJDET,DCJDET,PODET,ROODET
        COMMON /FI/FIH1,FIH2,FIH3,UXN,PXN,GXN,ROXN
       1   ,GIH
       2   ,FIH4,ZMDOTL,ZMDOTR
        COMMON /PRNT/ROX,PX,UX,ZX
C***********************************************************************
        ENTRY FLUXE1(L,I,MIN)
C***********************************************************************
        DT2=DT/2.
        GO TO (1,2,3,4,5,6),NFLUX
 1      CONTINUE
```

```
      DUDXIX=DUDXIL
      DRDXIX=DRDXIL
      DPDXIX=DPDXIL
      DGDXIX=0.5*GL*(DPDXIL/PL+DRDXIL/ROL)
      DUDTX=-DPDXIL
      DRODTX=-ROL**2*DUDXIL
      DPDTX=-GL**2*DUDXIL
       DRODTX=DRODTX-RAT*ROL*UL
      DPDTX=DRODTX*CL**2
      DGDTX=G6*GL*DPDTX/PL
      UX=UL
      PX=PL
      ROX=ROL
      ZX=ZL
      GX=GL
      GO TO 9
 6    CONTINUE
      DUDXIX=DUDXIR
      DPDXIX=DPDXIR
      DRDXIX=DRDXIR
      DGDXIX=0.5*GR*(DPDXIR/PR+DRDXIR/ROR)
      DUDTX=-DPDXIR
      DPDTX=-GR**2*DUDXIR
      DRODTX=-ROR**2*DUDXIR
      DRODTX=DRODTX-RAT*ROR*UR
      DPDTX=DRODTX*CR**2
      DGDTX=G6*GR*DPDTX/PR
      UX=UR
      PX=PR
      ROX=ROR
      ZX=ZR
      GX=GR
      GO TO 9
 2    CONTINUE
      BETA0=(MU2*(UL/CL+G7))**(1./MU2)
      A1=DUDXIL+DPDXIL/GL
      A0=A1+(G3/GL)*(CL*DGDXIL-G4*DPDXIL)*(BETA0**G5-1.)
      EVER1=-((GAMA-1)*UL+2*CL)*(BETA0**G13-1.)/(GAMA-3.)
      EVER2=4.*CL*(BETA0**G14-1.)/(3.*GAMA-5.)
      EVER=(EVER1+EVER2)*RAT/ROL
      A0=(A0+EVER)
      DUDAX=A0
      DPDAX=GL*BETA0*A0
      C0=MU2*(UL+G7*CL)
      UX=C0
      ROX=GL*BETA0/C0
      ZX=ZSTARL
      PX=ROX*C0**2/GAMA
      GX=ROX*C0
      DRODAX=ROL*BETA0**(1./G4)
     1          *((DRDXIL/ROL-DPDXIL/(GAMA*PL))*DSQRT(BETA0)
     2            +A0/C0)
      DGDAX=BETA0*DSQRT(BETA0)*(DGDXIL-G4*DPDXIL/CL)
     1     +G4*ROL*A0*BETA0**(1./G4)
      DPDAX=DPDAX+RAT*UX*C0*DSQRT(BETA0)
      G41=1./G4+0.5
      DRODAX=(DRDXIL-DPDXIL/(CL*CL))   *BETA0**G41+DPDAX/(C0*C0)
      DGDAX=0.5*GAMA*(PX*DRODAX+ROX*DPDAX)/GX
      DUDBX=-CL*BETA0**(-1./G4)/G4
      DPDBX=PL*BETA0**MU2/G6
      DRODBX=ROL*BETA0**(-MU2)/G4
      DGDBX=GL
      GO TO 9
 5    CONTINUE
      BETA0=(MU2*(-UR/CR+G7))**(1./MU2)
      A1=DUDXIR-DPDXIR/GR
      A0=A1+(G3/GR)*(-CR*DGDXIR+G4*DPDXIR)*(BETA0**G5-1.)
      EVER1=(-(GAMA-1.)*UR+2*CR)*(BETA0**G13-1.)/(GAMA-3.)
      EVER2=-4.*CR*(BETA0**G14-1.)/(3.*GAMA-5.)
      EVER=(EVER1+EVER2)*RAT/ROR
      A0=(A0+EVER)
      DUDAX=A0
```

```
      DPDAX=-GR*BETA0*A0
      C0=MU2*(-UR+G7*CR)
      DRODAX=-ROR*BETA0**(1./G4)
     1          *((-DRDXIR/ROR+DPDXIR/(GAMA*PR))*DSQRT(BETA0)
     2            +A0/C0)
      UX=-C0
      ROX=GR*BETA0/C0
      ZX=ZSTARR
      PX=ROX*C0**2/GAMA
      GX=ROX*C0
      DGDAX=BETA0*DSQRT(BETA0)*(-DGDXIR+G4*DPDXIR/CR)
     1     +G4*ROR*A0*BETA0**(1./G4)
      DGDAX=-DGDAX
      DPDAX=DPDAX-RAT*UX*C0*DSQRT(BETA0)
         G41=1./G4+0.5
      DRODAX=(DRDXIR-DPDXIR/(CR*CR))   *BETA0**G41+DPDAX/(C0*C0)
      DGDAX=0.5*GAMA*(PX*DRODAX+ROX*DPDAX)/GX
      DUDBX=CR*BETA0**(-1./G4)/G4
      DPDBX=PR*BETA0**MU2/G6
      DRODBX=ROR*BETA0**(-MU2)/G4
      DGDBX=GR
      GO TO 9
3     CONTINUE
      UX=USTAR
      PX=PSTAR
      ROX=RSTARL
      ZX=ZSTARL
      GX=GSTARL
      DUDXIX=-DPIDT/GSTARL**2
      DPDXIX=-DUIDT
      DUDXIX=DUDXIX-RAT*USTAR/RSTARL
      IF (.NOT.HELEML) GO TO 32
31    CONTINUE
      DRDXIX=(RSTARL/WL)**2*(3.*DUIDT+DPIDT*(1.+3.*(WL/GSTARL)**2)/WL
     1                       +DUDXIL*WL*((GL/WL)**2+3.)+3.*DPDXIL
     2                       +DRDXIL*(WL/ROL)**2)
      EVER1=UL*RSTARL**2*RAT*((GL/WL)**2+1.)/(ROL*WL)
      EVER2=2.*RSTARL*USTAR*RAT/WL
      DRDXIX=DRDXIX+EVER1+EVER2
      GO TO 33
32    CONTINUE
      BETA=GSTARL/GL
      SQB=DSQRT(BETA)
      DRODA=ROL*BETA**(1./G4)*(SQB*(DRDXIL/ROL-DPDXIL/(GAMA*PL))
     1                       +ASTARL/CSTARL)
      DRDXIX=DRODA/SQB+DPIDT/(GSTARL*CSTARL**2)
      DPDA=GSTARL*(ASTARL+RAT*USTAR*CSTARL/(GL*   SQB))
      G41=1./G4+0.5
      DRODA=(DRDXIL-DPDXIL/(CL*CL))   *BETA**G41+DPDA/(CSTARL**2)
      DRDXIX=   DRODA/SQB+DPIDT/(GSTARL*CSTARL**2)
33    CONTINUE
      DGDXIX=0.5*GX*(DPDXIX/PX+DRDXIX/ROX)
      DUDTX=DUIDT
      DPDTX=DPIDT
      DGDTX=G6*GSTARL*DPIDT/PSTAR
      DRODTX=DPIDT/CSTARL**2
      GO TO 9
4     CONTINUE
      UX=USTAR
      PX=PSTAR
      ROX=RSTARR
      ZX=ZSTARR
      GX=GSTARR
      DUDXIX=-DPIDT/GSTARR**2
      DUDXIX=DUDXIX-RAT*USTAR/RSTARR
      DPDXIX=-DUIDT
      IF (.NOT.HELEMR) GO TO 42
41    CONTINUE
      DRDXIX=(RSTARR/WR)**2*(3.*DUIDT-DPIDT*(1.+3.*(WR/GSTARR)**2)/WR
     1                       -DUDXIR*WR*((GR/WR)**2+3.)+3.*DPDXIR
     2                       +DRDXIR*(WR/ROR)**2)
      EVER1=UR*RSTARR**2*RAT*((GR/WR)**2+1.)/(ROR*WR)
```

```
      EVER2=2.*RSTARR*USTAR*RAT/WR
      DRDXIX=DRDXIX-EVER1-EVER2
      GO TO 43
   42 CONTINUE
      BETA=GSTARR/GR
      SQB=DSQRT(BETA)
      DRODA=-ROR*BETA**(1./G4)*(SQB*(-DRDXIR/ROR+DPDXIR/(GAMA*PR))
     1                         +ASTARR/CSTARR)
      DRDXIX=DRODA/SQB-DPIDT/(GSTARR*CSTARR**2)
      DPDA=-GSTARR*(ASTARR+RAT*USTAR*CSTARR/(GR*   SQB))
      G41=1./G4+0.5
      DRODA=(DRDXIR-DPDXIR/(CR*CR))   *BETA**G41+DPDA/(CSTARR**2)
      DRDXIX=   DRODA/SQB-DPIDT/(GSTARR*CSTARR**2)
   43 CONTINUE
      DGDXIX=0.5*GX*(DPDXIX/PX+DRDXIX/ROX)
      DUDTX=DUIDT
      DPDTX=DPIDT
      DGDTX=G6*GSTARR*DPIDT/PSTAR
      DRODTX=DPIDT/CSTARR**2
      GO TO 9
    9 CONTINUE
      RETURN
C*********************************************************************
      ENTRY FLUXE2(L,I,MIN)
C*********************************************************************
      FI1=ROX*UX
      FI2=ROX*UX**2+PX
      FI2=FI2-PX
      FI3=UX*(G12*PX+0.5*ROX*UX**2+ZX*ROX*QDET)
      FI4=ZX*ROX*UX
      ROU00=ROX*UX
      GO TO(10,20,30,40,50,60), NFLUX
   10 CONTINUE
   60 CONTINUE
      DFDXI1=DRDXIX*UX+ROX*DUDXIX
      DFDXI2=DRDXIX*UX**2+2.*ROX*UX*DUDXIX+DPDXIX
      DFDXI2=DFDXI2-DPDXIX
      DFDXI3=DUDXIX*(G12*PX+0.5*ROX*UX**2)
     1      +UX*(G12*DPDXIX+0.5*DRDXIX*UX**2+ROX*UX*DUDXIX)
      DFIDT1=DRODTX*UX+ROX*DUDTX
      DFIDT2=DRODTX*UX**2+2.*ROX*UX*DUDTX+DPDTX
      DFIDT2=DFIDT2-DPDTX
      DFIDT3=DUDTX*(G12*PX+0.5*ROX*UX**2)
     1      +UX*(G12*DPDTX+0.5*DRODTX*UX**2+ROX*UX*DUDTX)
      FIDOT1=-ROU00*DFDXI1+DFIDT1
      FIDOT2=-ROU00*DFDXI2+DFIDT2
      FIDOT3=-ROU00*DFDXI3+DFIDT3
      UXDOT=-ROU00*DUDXIX+DUDTX
      PXDOT=-ROU00*DPDXIX+DPDTX
      GXDOT=-ROU00*DGDXIX+DGDTX
      ROXDOT=-ROU00*DRDXIX+DRODTX
      FIH1=FI1+DT2*FIDOT1
      FIH2=FI2+DT2*FIDOT2
      GIH=PX+DT2*PXDOT
      FIH3=FI3+DT2*FIDOT3
      FIH4=FI4
      UXN=UX+DT*UXDOT
      PXN=PX+DT*PXDOT
      GXN=GX+DT*GXDOT
      ROXN=ROX+DT*ROXDOT
      GO TO 90
   20 CONTINUE
      EV0=GL*DSQRT(BETA0)
  201 CONTINUE
      DFIDA1=DRODAX*UX+ROX*DUDAX
      DFIDA2=DRODAX*UX**2+2.*ROX*UX*DUDAX+DPDAX
      DFIDA2=DFIDA2-DPDAX
      DFIDA3=DUDAX*(G12*PX+0.5*ROX*UX**2)
     1       +UX*(G12*DPDAX+0.5*DRODAX*UX**2+ROX*UX*DUDAX)
      FIDOT1=-EV0*DFIDA1
      FIDOT2=-EV0*DFIDA2
      FIDOT3=-EV0*DFIDA3
```

```
      FIH1=FI1+DT2*FIDOT1
      FIH2=FI2+DT2*FIDOT2
      FIH3=FI3+DT2*FIDOT3
      FIH4=FI4
      GA=DGDAX
      IF(NFLUX.EQ.5)GA=-GA
      DROUA=UX*DRODAX+ROX*DUDAX
      BETAPR=0.5*DSQRT(BETA0)*(GA-DROUA)
      FIH2=FIH2-DPDBX*BETAPR*DT2
      UXDOT=-EV0*DUDAX+BETAPR*DUDBX
      PXDOT=-EV0*DPDAX+BETAPR*DPDBX
      GIH=PX+DT2*PXDOT
      GXDOT=-EV0*DGDAX+BETAPR*DGDBX
      ROXDOT=-EV0*DRODAX+BETAPR*DRODBX
      UXN=UX+DT*UXDOT
      PXN=PX+DT*PXDOT
      GXN=GX+DT*GXDOT
      ROXN=ROX+DT*ROXDOT
      GO TO 90
   50 CONTINUE
      EV0=-GR*DSQRT(BETA0)
      GO TO 201
   30 CONTINUE
   40 CONTINUE
      GO TO 60
   90 CONTINUE
      RETURN
      END
```

## 6. DISTRIBUTION LIST

No. of Copies

1. Defense Technical Information Center
   Cameron Station
   Alexandria, VA 22314 . . . . . . . . . . . . . . . . . . . . . . . . 2

2. Library, Code 0142
   Naval Postgraduate School
   Monterey, CA 93943-5100. . . . . . . . . . . . . . . . . . . . . . . 2

3. Department Chairman, Code 67
   Department of Aeronautics
   Naval Postgraduate School
   Monterey, CA 93943-5100. . . . . . . . . . . . . . . . . . . . . . . 1

4. Distinguished Professor Allen E. Fuhs
   Space Systems Academic Group, Code 72
   Naval Postgraduate School
   Monterey, CA 93943-5100. . . . . . . . . . . . . . . . . . . . . . . 5

5. Dr. Neil Griff
   SDIO/DEO
   Washington, DC 20301-7100. . . . . . . . . . . . . . . . . . . . . . 3

6. Mr. Bruce Pierce
   SDIO/DEO
   Washington, DC 20301-7100. . . . . . . . . . . . . . . . . . . . . . 1

7. Dr. Joseph Falcovitz
   Code 72
   Naval Postgraduate School
   Monterey, CA 93943-5100. . . . . . . . . . . . . . . . . . . . . . . 5

No. of Copies

8. Professor Oscar Biblarz
Department of Aeronautics, Code 67
Naval Postgraduate School
Monterey, CA 93943-5100. . . . . . . . . . . . . . . . . . . . . . . 1

9. Research Administration Office
Code 012
Naval Postgraduate School
Monterey, CA 93943-5100. . . . . . . . . . . . . . . . . . . . . . . 1

10. Dr. P. Avizonis
Air Force Weapons Laboratory
Kirtland Air Force Base, NM 87117 . . . . . . . . . . . . . . . . . . 1

11. Dr. John Lawless
Space Power Inc.
1977 Concourse Drive
San Jose, CA 95131 . . . . . . . . . . . . . . . . . . . . . . . . 1

12. Dr. Mark Thornton
Boeing Aerospace Company
Post Office Box 3999
Seattle, WA 98124-2499 . . . . . . . . . . . . . . . . . . . . . . 1

13. LT. Mark Price
AFRPL
Edwards AFB, CA 93523 . . . . . . . . . . . . . . . . . . . . . . 1

14. Mr. Arthur W. Rogers
Space Systems Division
Hughes Aircraft Co.
P. O. Box 92919, Los Angeles, CA 90009 . . . . . . . . . . . . . . . 1

No. of Copies

15. LCOL Rick Babcock, USAF
Air Force Geophysical Laboratory
Hanscomb Field
Bedford, MA 01730 . . . . . . . . . . . . . . . . . . . . . . . . 1

16. Mr. Ronald J. Hoffman
Plume Technology and Spacecraft
Contamination Division
Science Applications International Co.
10000 Santa Monica Blvd., Suite 320
Los Angeles, CA 90067 . . . . . . . . . . . . . . . . . . . . . . . 1